



Víctor Murciano Durán

Primavera, 2018 - 2019 Q2

Grado en Ingeniería Informática

TRABAJO DE FIN DE GRADO

A process-aware tool for the worldwide request and distribution of medical
supplies for Chagas

Director: BESIM BILALLI

Co-Director: ALBERTO ABELLO GAMAZO

Agradecimientos

Antes que nada quisiera agradecer a mis directores Besim y Alberto por darme la oportunidad de ser parte de este proyecto y de confiar y guiarme durante todos estos meses.

También me gustaría dar un agradecimiento especial a los representantes de la OMS que me han ayudado y apoyado siempre que han podido.

Además me gustaría agradecer a mi familia, por apoyarme siempre en todo momento y darme ánimos a seguir.

Y por último me gustaría agradecer a los amigos que me dieron apoyo moral y consejos durante estos meses.

Resumen

Las enfermedades tropicales desatendidas (ETD) son un grupo de enfermedades transmisibles que prevalecen en condiciones tropicales en 149 países. Afectan principalmente a las poblaciones que viven en la pobreza, sin saneamiento adecuado y en estrecho contacto con insectos u otros tipos de animales que pueden portar estas enfermedades.

Estos medicamentos no tienen un coste muy alto, pero para estos países que viven en pobreza, conseguirlos es muy difícil. La OMS es uno de los principales proveedores de estos medicamentos y recae en ellos la capacidad de distribuirlos para poder ayudar a estos países e intentar conseguir la erradicación de estas enfermedades.

Este proyecto pertenece a uno mayor y dentro de este, es el segundo, siendo la continuación directa del primer proyecto realizado, en aquel se desarrolló un sistema para las provisiones de medicamentos. En este se ha construido una aplicación para permitir a las diferentes instituciones repartidas por el mundo que puedan solicitar estos medicamentos a la OMS y de esta forma sustituir el proceso rudimentario que había hasta hoy en día. Al principio, el proyecto estaba destinado a ser usado para la enfermedad de Chagas, sin embargo, debido a los aspectos comunes en el proceso, se espera que también sea utilizado por otras ETD.

Los diferentes procesos implementados dentro de este proyecto incluyen el “Request Process” y “Manage Temporal Institutions” entre otros, pero principalmente está todo concentrado en el primero. Además, la tesis está compuesta de dos tareas principales: la preparación del entorno y aprendizaje de estas herramientas y el desarrollo.

Debido a la naturaleza del problema, se eligió una herramienta de BPM para todos los proyectos, en este caso, Bonita BPM. Durante el desarrollo se tuvo que formalizar el proceso para pedir medicamentos para de esta forma poder traspasarlo a un proceso digital con BPM, además, durante la creación de este, debido a las limitaciones de Bonita BPM, se vio que para la creación de usuarios se debía hacer desde el exterior mediante llamadas API.

Resum

Les malalties tropicals desateses (MTD) són un grup de malalties transmissibles que prevalen en condicions tropicals a 149 països. Afecten principalment a les poblacions que viuen en la pobresa, sense sanejament adequat i en estret contacte amb insectes o altres tipus d'animals que poden portar aquestes malalties.

Aquests medicaments no tenen un cost molt alt, però per a aquests països que viuen en pobresa, aconseguir-los és molt difícil. L'OMS és un dels principals proveïdors d'aquests medicaments i recau en ells la capacitat de distribuir-los per poder ajudar a aquests països i intentar aconseguir l'eradicació d'aquestes malalties.

Aquest projecte pertany a un major i dins d'aquest, és el segon, sent la continuació directa del primer projecte realitzat, en aquell es va desenvolupar un sistema per a les previsions de medicaments. En aquest s'ha construït un aplicació per permetre a les diferents institucions repartides pel món que puguin sol·licitar aquests medicaments a l'OMS i d'aquesta manera substituir el procés rudimentari que hi havia fins avui dia. Al principi, el projecte estava destinat a ser usat per a la malaltia de Chagas, però, a causa dels aspectes comuns en el procés, s'espera que també sigui utilitzat per altres MTD.

Els diferents processos implementats dins d'aquest projecte inclouen el "Request Process" i "Manage Temporal Institutions" entre d'altres, però principalment està tot concentrat en el primer. A més, la tesi està composta de dues tasques principals: la preparació de l'entorn i aprenentatge d'aquestes eines i el desenvolupament.

A causa de la naturalesa del problema, es va triar una eina de BPM per a tots els projectes, en aquest cas, Bonita BPM. Durant el desenvolupament es va haver de formalitzar el procés per demanar medicaments per d'aquesta manera poder traspasar-lo a un procés digital amb BPM, a més, durant la creació d'aquest, a causa de les limitacions de Bonita BPM, es va veure que per a la creació d'usuaris s'havia de fer desde el exterior mitjançant trucades API.

Abstract

Neglected tropical diseases (NTD) are a group of communicable diseases that prevail in tropical conditions in 149 countries. They mainly affect populations living in poverty, without adequate sanitation and in close contact with insects or other types of animals that can carry these diseases.

These medicines do not have a very high cost, but for the countries that live in poverty, getting them is very difficult. WHO is one of the main providers of these medicines and holds the responsibility to distribute them in order to help the countries and to try to eradicate these diseases.

This project belongs to a larger one and within this, it is the second one, being the direct continuation of the first project carried out, in that a system for the predictions of medicines was developed. In this an application has been built to allow the different institutions around the world to request medicines from the WHO and in this way replace the rudimentary process that had been present up to nowadays. In the beginning the project was meant to be used for the Chagas disease, however, due to commonalities in the process, it is expected to be used by other NTDs too.

The different processes implemented within this project include the “Request Process” and “Manage Temporal Institutions” among others, but mainly it is concentrated in the first one. In addition, the thesis is composed of two main tasks: the preparation of the environment and learning of the tools and development.

Due to the nature of the problem, a BPM tool was chosen for all the projects, in this case, Bonita BPM. During the development “Request Process” had to be formalized to be able to be transferred to a digital process with BPM, also, during the creation of this, due to the limitations of Bonita BPM, the creation of users was required to be done through an external form using API calls.

Índice general

1. Contexto	1
1.1. Introducción	1
1.2. Stakeholders	3
1.3. Estado del arte	4
1.3.1. Ventajas e inconvenientes del BPM	5
1.3.2. Alternativas Existentes	6
1.3.3. Aplicaciones Similares	7
1.3.4. Conclusiones sobre el estado del arte	8
1.4. Objetivos	8
2. Alcance y Metodología	10
2.1. Alcance	10
2.2. Obstáculos	11
2.3. Metodología y herramientas	12
2.3.1. Métodos de trabajo y herramientas para su seguimiento	12
2.3.2. Métodos de validación y seguimiento	15
2.4. Conclusiones	16
3. Planificación temporal	17
3.1. Calendario	17
3.2. Descripción de los objetivos y tareas a realizar	18
3.2.1. Inicio y planificación	18
3.2.2. Desarrollo	18

3.2.3. Fase final	19
3.2.4. Tiempo Estimado	20
3.3. Recursos	20
3.4. Valoración de alternativas y plan de acción	22
3.5. Diagrama de Gantt	23
4. Gestión Económica	24
4.1. Identificación y estimación de los costes	24
4.1.1. Recursos humanos	24
4.1.2. Recursos software/hardware	26
4.1.3. Otros	26
4.1.4. Imprevistos y plan de contingencia	27
4.1.5. Costes totales	28
4.2. Control de Gestión	28
5. Sostenibilidad y compromiso social	30
5.1. Autoevaluación	30
5.2. Económica	31
5.3. Social	31
5.4. Ambiental	32
5.5. Matriz de sostenibilidad	32
6. Bonita BPM	33
6.1. ¿Qué es Bonita BPM?	33
6.2. Bonita BPM Studio	34
6.3. Bonita BPM Platform	37
6.3.1. UI Designer	37
6.3.2. Bonita User Experience	40
6.3.3. Bonita Engine	42
6.4. Otros Requerimientos de Bonita	43

7. Desarrollo del proyecto	44
7.1. ¿Qué vamos a adaptar?	44
7.2. Actores y roles del proyecto	46
7.3. Registro de usuarios	47
7.4. Objetos de negocio "Business Objects"	50
7.5. Procesos diseñados	51
7.6. Procesos principales	52
7.6.1. Peticion de medicamentos "Reques Process"	52
7.6.2. Enviar email "Send Email"	62
7.6.3. Gestión de las instituciones temporales "Manage Temporal Institution"	63
7.7. Procesos secundarios	64
7.7.1. Proceso para revisar los pedidos "See Requests"	64
7.7.2. Proceso para editar el perfil "Edit Profile"	65
7.7.3. Proceso para cambiar la contraseña "Change Password"	66
7.8. Aplicaciones	67
7.8.1. Aplicación para la petición de los usuarios	67
7.8.2. Aplicación para la reasignación de las instituciones temporales	68
7.8.3. Aplicación para la consulta de todas las peticiones	68
7.9. Testeo	69
7.10.Despliegue del servidor	70
8. Conclusiones	71
A. Símbolos BPM	73
A.1. Gateways	73
A.1.1. Parallel gateway	73
A.1.2. Exclusive gateway	73
A.1.3. Inclusive gateway	74
A.2. Tasks	74
A.2.1. Human task	74
A.2.2. Service task	75

A.2.3. Task with connector	75
A.3. Activities	75
A.3.1. Call activity	75
A.4. Events	76
A.4.1. Start event	76
A.4.2. End event	76
A.4.3. Non-interrupting timer boundary event	76
B. Competencias Técnicas de Ingeniería del Software	77
B.1. Competencias Técnicas	77
Bibliografía	79

Capítulo 1

Contexto

1.1. Introducción

Hoy en día vemos que casi todo se mueve por ordenador o todo está controlado mediante tecnología, esto es así hasta tal punto que casi todos los procesos que se realizan en empresas están supervisados y realizados de alguna forma por alguna máquina mediante una aplicación, y la misma comunicación entre las tareas se realiza a través de ella de forma automática. Los tiempos en que todo se realizaba a través de mensajería física o mediante llamadas telefónicas cada vez va a menos, se busca la necesidad de poder realizar procesos sin necesidad de una comunicación constante entre 2 personas y conseguir una automatización completa de todo el proceso, además de saber en todo momento en que punto está para poder tener un control total sobre él. En esta última parte entraría el proyecto a realizar. Para conseguirlo haremos uso del Business Process Management BPM, este se puede definir como una disciplina o nueva categoría de software empresarial que permite a las empresas modelar, implementar y ejecutar conjuntos de actividades interrelacionadas –es decir, Procesos– de cualquier naturaleza, tanto dentro de un departamento como permeando la entidad en su conjunto, con extensiones para incluir a los clientes, proveedores y otros agentes como participantes en las tareas de los procesos. Según la web de *AuraPortal*¹, una empresa dedicada al uso de BPM, estos aseguran que la eficiencia mejora en todo, aumenta la productividad y se reducen los costes de tiempo entre otras cosas. De esta forma, con todo esto último que hemos dicho, podremos saber en todo momento la

¹<https://www.auraportal.com>

situación exacta donde nuestro proceso esta parado y a quien le corresponde la obligación de realizar la tarea en concreto para, si es necesario, poder agilizar el proceso ayudándole o avisándole de que debe realizarla. El proyecto que se ha desarrollado es una colaboración con la OMS (Organización mundial de la salud) y el grupo de investigación Database Technologies and Information Management (DTIM) que es parte del grupo ESSI de la FIB (Facultad de Informática de Barcelona). En este, el objetivo es el de conseguir mediante BPM la construcción de una app web en la cual tengamos una serie de herramientas donde la más importante sea la que nos permita pedir medicinas mediante un proceso llamado 'Request Process' a la OMS sin tener que hacerlo a través de la forma tradicional, las otras herramientas o procesos también se realizarán como BPM. Además cabe añadir y recalcar que este proyecto pertenece a otro más grande, en la siguiente imagen se puede ver un esquema completo a alto nivel de este y marcado en rojo lo que se ha desarrollado en el proyecto:

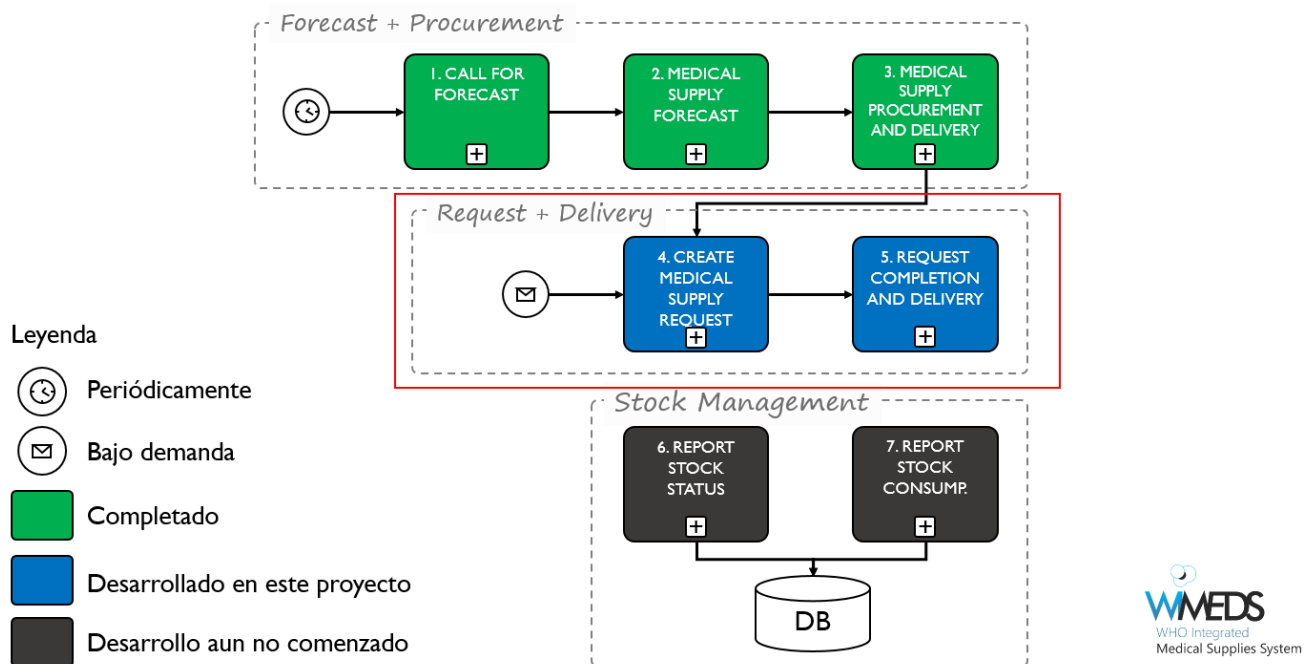


Figura 1.1: Esquema general del proyecto

Durante el documento, cuando se hable sobre el anterior proyecto, será lo ya completado que vemos en verde.

1.2. Stakeholders

En el proyecto hay varias partes interesadas en la realización de este mismo, estas son las siguientes:

Director del proyecto

El encargado de este rol es el que ayuda y dirige al desarrollador a realizar este proyecto en todo momento tanto en el proceso de diseño como en el desarrollo de la *app*² siguiendo la estructura BPM. Además junto al desarrollador se pondrá en contacto con la OMS mediante reuniones para saber sus necesidades con respecto a la parte que les corresponde. En este caso, este rol le pertenece a Besim Billali.

Desarrollador

Es el que se encarga de diseñar y de desarrollar la app mediante BPM, para todo este proceso, tal y como se ha dicho justo antes, tendrá la ayuda del director que le guiará en todo lo que necesite. Además se encarga de hacer las pruebas necesarias para asegurarse de que todo el software desarrollado funcione perfectamente y sin problemas. También es el autor del proyecto ya que es su TFG en este caso.

OMS

La OMS es una de las partes más interesadas con respecto al final del proyecto, serán ellos los que harán uso de este software para poder gestionar de manera mucho más eficiente las peticiones y reparto de medicinas hacía distintas instituciones y diferentes países, así dejarán de usar el método tradicional para todo este tipo de cosas. Ellos deberán indicarle al director y al desarrollador sus necesidades para que se puedan aplicar y verse reflejadas en la propia app al final de su desarrollo cuando se de uso.

²Aplicación

Usuarios finales de la aplicación

Por último, estos son los que utilizarán la aplicación tanto para pedir medicamentos como para consultar los diferentes pedidos que se hayan realizado. Aquí entrarían tanto los usuarios de diferentes países que piden medicamentos como los trabajadores de la OMS que se encarguen de gestionar todos los pedidos. El diseño de la aplicación debe ser lo más útil posible para que todos la puedan entender.



































1.3. Estado del arte


Hoy en día el BPM es una disciplina que se está utilizando cada vez más pero que las empresas tienen un poco de miedo a aplicarla debido a que implica un gran compromiso con todos sus usuarios, todo los procesos siguen unas directrices y de ahí no se pueden salir, por lo tanto pueden llegar a darse casos un tanto extraños cuando se realizan según que cosas de forma inadecuada.


Esta disciplina va a ser esencial en este proyecto ya que hoy en día para realizar todo este proceso de pedir medicamentos a la OMS, se realizaba mediante un envío de unos documentos, llamadas y mensajes, esto, como es normal, provocaba que no fuese fluido y que se produjesen retrasos de varios días por la falta de consistencia en el proceso y de no saber a veces en que punto se encontraba este. Por lo tanto el desarrollo de esta herramienta para realizar estos pedidos es primordial y sería un avance muy positivo no solo para la OMS sino para el mundo entero sobre todos en los países más necesitados.


Para decidir la herramienta a utilizar en el desarrollo con BPM, en el anterior proyecto a este, se hizo una comparativa entre varias herramientas y al final se hizo una gran comparación entre dos, estas fueron BonitaSoft y Flowable, con tal de decidir cual utilizar, se hizo una tabla (ver Figura 1.2) para ver realmente que ofrecía cada una y decidir cual quedarnos, esta tabla es la siguiente:

Como podemos ver, BonitaSoft, era mejor en la gran mayoría de las cosas.

BPM System	Bonitasoft	Flowable	Notes
"Zero-Coding" approach			Bonitasoft no necesita programar mucho, la mayoría de cosas están dadas
Connectivity			Bonitasoft proporciona conectores fácilmente implementados (por ejemplo, conectores para más bases de datos)
Form designer			Hay widgets que faltan en flowable (e.g., tables)
Working environment			Bonitasoft tiene una aplicación de escritorio (i.e., Studio) para el desarrollo, Flowable en cambio tiene una basada en web (i.e., Modeler)
Installation			Los dos tienen manuales y guías de instalación
System behavior			En caso de parar, eliminar o actualizar un proceso su comportamiento es similar.
Actors & Roles			Bonitasoft permite crear más relaciones entre actores y roles (e.g., Groups, Memberships)
Different profiles			Ninguno permite crear un perfil diferente de Usuario o Administrador (e.g., Supervisor)
Process analysis			Ninguno tiene una herramienta para analizar el flujo de los procesos (e.g., simulations)
Visualization			Flowable permite ver el flujo del proceso en ejecución.
Community			Bonitasoft community reacciona a los cambios en menor medida que Flowable community
Documentation			Los dos proporcionan buena documentación
Process data			Los dos proporcionan los datos que pertenecen a los procesos (i.e., metadata)
Low learning curve			Bonitasoft es más fácil de aprender
Maturity			Flowable es muy nuevo
Data			Bonitasoft permite crear los Business Objects desde el Studio
Open source			Flowable es completamente open source, en cambio Bonitasoft tiene su subscription edition.


Bueno


Medio


Poco



No

Figura 1.2: Tabla de comparación. Creada por Clara Gaset en su trabajo [20].

1.3.1. Ventajas e inconvenientes del BPM

Para ver realmente lo que ofrece BPM, a continuación se muestran sus ventajas e inconvenientes:

Ventajas:

- **Agilidad:** BPM tiene una gran capacidad de responder a oportunidades o amenazas en tiempo real. La agilidad también permite que las empresas salgan al mercado más rápido. De esta manera, los clientes reciben productos rápidamente y las empresas pueden utilizar los comentarios de los usuarios finales para el siguiente producto o actualización. Dado que los clientes ahora exigen constantemente una mayor rapidez esto es un gran beneficio para todas las empresas.
- **Transparencia:** se muestra y se entiende cómo funciona un proceso, incluidos cada paso necesario, la tecnología utilizada y cada persona involucrada. La transparencia se puede

habilitar para las partes interesadas hasta los trabajadores. También aumenta la comunicación, la productividad y la eficiencia entre los trabajadores.

- Eficiencia: hacer más por menos. Menos, en este caso, pueden ser fondos o recursos. Las empresas desean crear más productos sin triplicar sus gastos. BPM puede eliminar el desperdicio de recursos e identificar los mejores métodos para utilizar la financiación para el desarrollo y la distribución de productos.

Y en cuanto a los inconvenientes:

- Si BPM no se implementa correctamente puede ser una pérdida de tiempo y fondos.
- Muchas veces falla una implementación de BPM debido a la falta de comunicación. BPM requiere análisis, monitoreo y cambios tecnológicos.
- Los cambios afectan tanto a los trabajadores como a los clientes. Y si los trabajadores no se incorporan temprano a la integración de BPM, puede llevar al fracaso. Lo que puede llevar a la pérdida de beneficios y tiempo.
- Además, no implementar un sistema BPM con características clave (soporte móvil, modelado de procesos, etc.) puede afectar las soluciones propuestas para un proceso.

Como se puede observar hay un gran beneficio, pero hubo que tener muy en cuenta los inconvenientes para evitar que el proyecto se fuera a nada.

1.3.2. Alternativas Existentes

Después de indagar un tiempo buscando aplicaciones alternativas a la que queremos crear, se llegó a la conclusión de que no había ninguna que se utilizase para esto. Como se ha comentado antes, lo único que se ha usado como referencia y en lo que nos basamos para hacer la aplicación es en todo el proceso que se hacía antes para poder pedir estos medicamentos, es decir se ha intentado pasar todas esas fases que antes se hacían de forma manual y lenta a la aplicación y de esta forma gracias a BPM agilizamos y organizamos todo el proceso.

1.3.3. Aplicaciones Similares

Con respecto a aplicaciones similares existen unas llamada 'My Social Reader y Wehey' (ver Figuras 1.3 y 1.4) creadas por la empresa "BPM Social media", estas son unas de las muchas cosas que han hecho, pero nos fijamos en ellas porque aunque no son en web, son las únicas aplicaciones que han hecho haciendo uso del BPM. Por otro lado, muchos bancos, hoy en día, tienen aplicaciones BPM y casi todos los procesos los hacen siguiendo esta forma de desarrollo, eso si, con respecto a aplicaciones web es difícil encontrarlas. En la web de BonitaSoft, en el apartado de *success stories*³ se pueden encontrar una serie de compañías que han realizado diferentes proyectos con "Bonita BPM", al ver una gran cantidad de proyectos ha reforzado aun más la idea de hacerlo mediante esta herramienta.

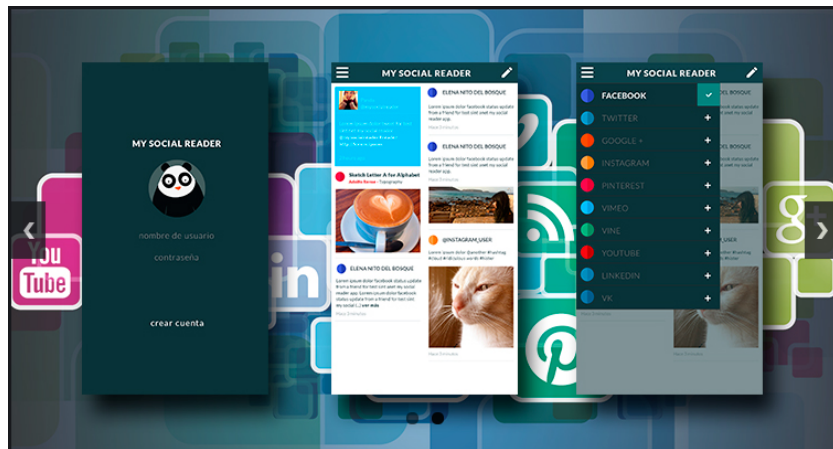


Figura 1.3: My Social Reader. <https://www.bpmsocialmedia.com/>

Pero, hay una en la que si que nos hemos fijado y que es muy similar a la que se ha desarrollado, esto es debido a que es el anterior proyecto. El objetivo de aquel era hacer las previsiones de los medicamentos que se van a pedir por países mediante una app web como la que se ha realizado en este. Por lo tanto con este último si que se pudo hacer una idea de como iba a ser porque se hizo de la misma forma, mediante "Bonita BPM" tal y como se ha hecho con el nuestro. Gracias a ello, no se tuvo la necesidad de reinventar la rueda otra vez y se aprendió de lo ya visto en el anterior.

³<https://www.bonitasoft.com/customer-stories>

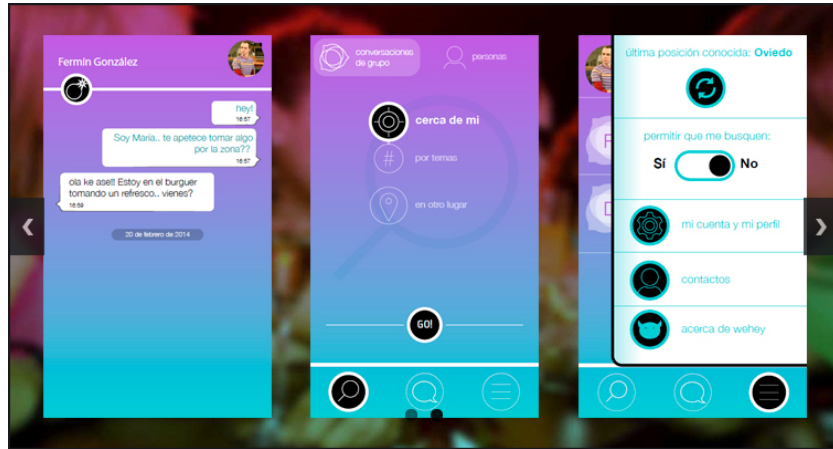


Figura 1.4: Wehey. <https://www.bpmsocialmedia.com/>

1.3.4. Conclusiones sobre el estado del arte

Como resumen de todo lo anterior dicho, básicamente se ha creado una aplicación nueva para poder optimizar el proceso de la solicitud de medicamentos, para ello se tuvo en cuenta todas las directrices que nos dijeron desde la OMS y se basó en el proceso que ya había anteriormente a esta, para de esta forma adaptarlo a una aplicación web. Además de esto, se han añadido todas las funcionalidades que son necesarias y que hemos visto que tienen sentido de cara al uso por parte de los usuarios.

También, como hemos dicho justo antes de esto, por como se ve la aplicación y algunas cosas que utiliza, se ha cogido como referencia el proyecto hecho antes que este, esto no quiere decir que se ha adaptado esa aplicación, solo se utilizan algunas de sus cosas como referencia ya que se ha hecho mediante las mismas herramientas. Y como hemos podido ver, si se hace con cuidado, hacerlo con BPM dará muchísimas ventajas.

Por lo tanto, la aplicación es completamente nueva y adapta y mejora el proceso manual que había antes, automatizando muchos pasos de este.

1.4. Objetivos

Una vez analizado todo el contexto, se puede hablar de los objetivos principales que se han intentado conseguir con este proyecto.

- El primero de todos y el principal es el de desarrollar una aplicación web en la que se pueda realizar el proceso de pedir medicinas, editar nuestro perfil de la aplicación por si fuese necesario y poder ver el estado de los pedidos realizados, además de otras funcionalidades que puedan surgir, todo esto por parte del usuario de los diferentes países que realizan la petición. Por otro lado, para los trabajadores de la OMS que deben gestionar las distintas peticiones, se les creará una serie de tareas para poder aceptar o rechazar las solicitudes y poder ayudar de alguna forma explicando a la persona que quiera pedir que cosas ha de modificar si hay algo erróneo, otra en la que podrán crear y poner toda la información del envío de los documentos y por último una en la que deben gestionar las nueva instituciones que los usuarios creen, más adelante se explicará esto.
- Segundo y no menos importante, se debe crear una base de datos consistente para poder manejar todos los datos esenciales para poder ejecutar el proceso satisfactoriamente y tener siempre todo lo necesario a mano, una mala implementación de esto puede hacer que nada funcione como debería.
- Por último, para realizar la petición de los medicamentos se necesita registrar a los usuarios en el portal de "Bonita BPM", pero para ello se hará desde una fuente externa debido a que para iniciar un proceso es necesario estar registrado y por lo tanto no se puede crear un proceso de registro. Así que hay que buscar la forma de realizarlo desde otra fuente.

Capítulo 2

Alcance del proyecto

2.1. Alcance

Una vez aclarado de que va el proyecto, podemos hacernos una idea e indicar cual será el alcance de este. A continuación lo veremos:

Aplicación web mediante BPM

La aplicación a desarrollar se realiza a través de la herramienta "Bonita BPM" (más adelante se hablará de ella, pero resumidamente con lo que se ha visto, es la que permite crearla utilizando la estrategia BPM). En esta, lo principal que se ha creado es un proceso para realizar los pedidos de medicamentos desde diferentes países. Este debe adaptarse a las necesidades de la OMS respecto a los pedidos según las enfermedades por las cuales se realizan. Además, para regular todo el trafico de personas que puedan realizar estas peticiones, "Bonita BPM" tiene una plataforma propia donde se debe registrar al usuario y así evitar que todo el mundo pueda acceder. Por ello, en la misma app se puede editar el perfil y además ver el estado de los pedidos realizados en todo momento.

Sistema de integración de datos

Para que todo funcione de la mejor manera posible y se pueda adaptar a todos los tipos de casos, se necesita un esquema y base de datos robusto para conseguir un funcionamiento ágil.

La propia herramienta de Bonita BPM además proporciona todo lo necesario para realizar un back-end¹ en condiciones.



Figura 2.1: Logo de "BonitaSoft".

2.2. Obstáculos

Tiempo

Unos de los principales obstáculos encontrados es el tiempo que se dispone para realizar el proyecto, este no es mucho y podía ser que no se pudiese acabar todo lo propuesto a tiempo. Pero como en todo proyecto hay unas fechas que se han de cumplir y por lo tanto se ha trabajado mucho para que todo pueda salir adelante.

Problemas con los datos

Un mal esquema de datos a utilizar desde un principio podía provocar que el sistema no funcionase correctamente debido a este mal uso y se tuviesen que rehacer muchas cosas. Por lo tanto este es uno de los factores más importantes a la hora de realizar el diseño para evitar todos los problemas que puedan darse.

Problemas en el registro de la app

"Bonita BPM" tiene un problema que se ha comentado antes, para realizar el registro se ha de hacer desde una parte externa a la propia plataforma por lo tanto para hacer una de las cosas más necesarias para el funcionamiento hay que hacerlo desde fuera mediante alguna forma que

¹[2]Capa de acceso a datos

se ha pensado en la fase de diseño, esto nos puede provocar perdidas de tiempo si no le damos la importancia que se merece. Eso si, para que todo quede consistente el usuario no ha de darse cuenta que esto esta fuera del propio sistema.

Estrategia BPM

A pesar de que BPM es considerado algo muy bueno, a veces se hace difícil su aplicación ya que es necesario que el proceso siga una serie de directrices y nada puede salir o hacerse diferente de como esta planeado, por lo tanto de vez en cuando se pueden dar situaciones un tanto raras y complejas de solucionar. Aun así, como hemos explicado y visto en la parte de ventajas e inconvenientes (1.3.1), cuando realmente funciona bien y sigue lo establecido se nota muchísimo. Por lo tanto el obstáculo que puede generar es la perdida de tiempo intentando evitar situaciones extraordinarias que puedan ocurrir.

Cambios de última hora

Una de las cosas que también pueden obstaculizar bastante el proceso son todos aquellos cambios que se deciden hacer hacia final debido a cosas que no convencen a algún stakeholder, estos al no avisarse con suficiente antelación pueden perjudicar mucho el avance del proyecto y este quedarse un poco estancado, por lo tanto es algo a tener muy en cuenta de cara al desarrollo.

2.3. Metodología y herramientas

2.3.1. Métodos de trabajo y herramientas para su seguimiento

Para realizar todo este proyecto se utiliza una metodología agile, esta consiste en una metodología para el desarrollo de proyectos que precisan de rapidez y flexibilidad, es una filosofía que supone una forma distinta de trabajar y de organizarse. De tal forma que cada proyecto se divide en pequeñas partes que tienen que completarse y entregarse en pocas semanas. El objetivo es desarrollar productos y servicios de calidad que respondan a las necesidades de unos clientes cuyas prioridades cambian a una velocidad cada vez mayor como justo se ha comentado en

algunos de los obstáculos que podemos encontrar. Cada parte a completar recibe el nombre de sprint y tiene una duración fija.

Por lo tanto se ha ido haciendo todo a la vez sin dejar nada de lado estableciendo una serie de prioridades según lo que se necesite para ese sprint en concreto. Cada mes se realizan reuniones con la gente de la OMS, así que los sprints son de un mes y hay que realizar todo lo establecido para ese en concreto. Cada dos meses la reunión se realiza en la FIB y por lo tanto el desarrollador asiste a esa además del director y ponente, en cambio la otra se hace en Ginebra y solo acudirán el director y ponente.

Para poder organizar y gestionar todo lo que se hace durante el sprint se utiliza la herramienta de trabajo “Redmine” (ver Figura 2.3), en esta se puede poner todo lo que hay que hacer y lo que se está haciendo dividido en diferentes subcategorías, además se pone el tiempo que hemos trabajado en cada cosa y el estado de la funcionalidad a hacer, de esta forma se establece un seguimiento de todo lo que se está trabajando y lo que se puede hacer o no. Todo lo que se vaya a hacer en el sprint, se pone como tareas a realizar en “Redmine”, y así, mediante una fecha final que podemos poner, se verá como va todo el trabajo que se ha realizado hasta ese momento en el sprint. Principalmente se ha elegido esta herramienta debido a su fácil funcionamiento y la gran capacidad que tiene además de ser de código libre.

Para desarrollar la app, se hace uso de la herramienta “Bonita BPM”. Esta facilitará la creación de procesos BPM permitiendo crear tanto el front-end² como el back-end a la vez (ver Figura 2.4). Esto da una gran adaptabilidad y facilidad en el uso de los datos y el manejo de ellos entre las 2 capas. Este front-end está formado por una aplicación de diseño llamada “UI-Designer” (ver Figura 2.2) que nos permite hacer muchísimas cosas gracias a su gran versatilidad. En lo que respecta al código, “Bonita BPM” lo guarda internamente y si se quiere compartir el proyecto hay que extraerlo todo en un tipo de archivo especial que solo maneja el propio programa.

Por último, para compartir todo el trabajo hecho en “Bonita”, documentos, datos y las decisiones tomadas en las reuniones, se utiliza un repositorio creado con SVN (mediante la herramienta Tortoise SVN). Esta es una herramienta de control de versiones open source³ basada en un

²Capa de presentación, lo que ve el usuario final

³término que denota que un producto incluye permiso para usar su código fuente, documentos de diseño o

repositorio cuyo funcionamiento se asemeja enormemente al de un sistema de ficheros. En esta todo el mundo que tenga acceso podrá compartir todo lo que fuese necesario y de esta forma guardar todo. Se ha elegido esta, debido a que es una de las más utilizadas y por lo tanto funciona muy bien y además es de código libre

Estas herramientas fueron seleccionadas en el anterior proyecto a este y se decidió seguir utilizándolas para mantener una coherencia entre todo.

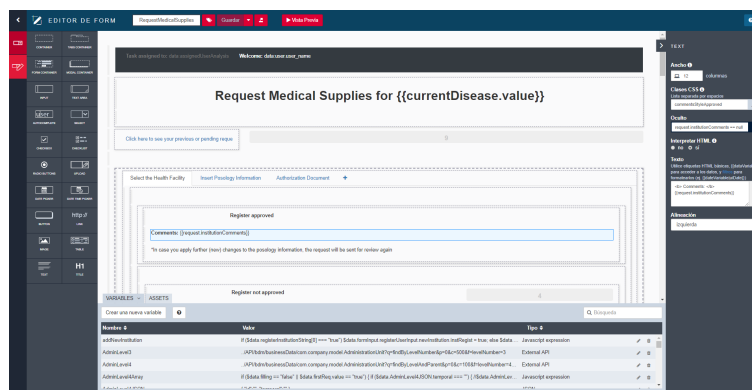


Figura 2.2: Captura del UI-Designer.

La base de datos integrada dentro de "Bonita" funciona mediante sentencias SQL.

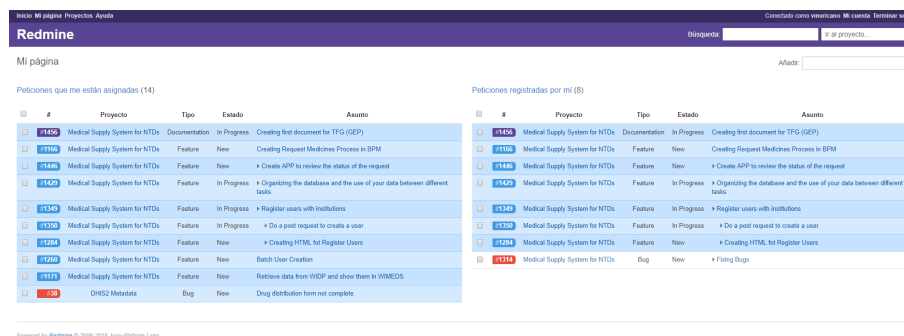


Figura 2.3: Captura de la página de un usuario en Redmine con sus tareas.

Por último, el registro en el portal de “Bonita” se realiza desde fuera gracias a que “Bonita” contenido

- **Reunión cada semana:** Cada semana el desarrollador se reúne con el director del proyecto para que le pueda dar “feedback” sobre el trabajo que está realizando, de esta forma, si hay algún problema, este se pueda solucionar cuanto antes y no arrastrarlo durante mucho tiempo. En esta reunión además, se habla sobre todo lo que hay que hacer en el sprint y sobre nuevas funcionalidades que se puedan añadir de cara a la demostración a realizar a final de mes. Si las dudas son pequeñas se utiliza el mail para una solución más rápida.

Si algo de lo que está planeado no llega a una de las reuniones, se tiene que incluir en lo pactado para la siguiente y realizar más horas si es necesario.

2.4. Conclusiones

Por último y para dar unas pequeñas conclusiones, es un trabajo al que hay que dedicarle muchas horas ya que se utilizan una serie de tecnologías bastante novedosas y que para adaptarse a ellas requieren un cierto tiempo. Pero aun así, creemos que es algo muy positivo y que en un futuro esto irá a más y más empresas adoptarán una estrategia como el BPM para el desarrollo de todos sus procesos gracias a las tantas cosas que puede ofrecer, como se ha ido explicando durante el documento.

Capítulo 3

Planificación temporal

3.1. Calendario

El proyecto tiene una duración aproximada de 4 meses y medio. Se marca como punto de inicio el primer día de clase presencial de GEP el 18 de febrero de 2019. La idea es acabar cuanto antes alrededor de unas 3 semanas antes de la entrega final para poder preparar la defensa del trabajo y solucionar todos los problemas que puedan surgir. El proyecto tiene una carga de trabajo de 350 horas + 75h de GEP, teniendo en cuenta la carga de horas por crédito (25 horas/crédito) y que el TFG son 15 créditos y GEP 3. Por lo tanto al estar hecho por solo una persona se tienen que hacer unas 28 horas semanales.

Además hace falta añadir que este TFG pertenece a un trabajo que se estaba realizando desde septiembre, y por lo tanto la fase inicial ya había pasado al empezar la fase de TFG en febrero. Aun así a continuación se explicarán las fases y se analizará como está distribuido todo. Por lo tanto la primera fase que se explicará de introducción se hizo antes de febrero, pero para explicarlo se va a suponer que esto se ha hecho unas semanas antes de GEP y en cuanto empieza este a la vez se entra de lleno en las iteraciones.

3.2. Descripción de los objetivos y tareas a realizar

3.2.1. Inicio y planificación

Como este proyecto es nuevo pero parte de unas herramientas dadas en el anterior proyecto como se ha explicado en el capítulo 2, al principio se ha realizado una trabajo de conocimiento e integración del funcionamiento de todas ellas, las primeras semanas se ha tratado de entender el problema a realizar y utilizar mediante demos la herramienta principal Bonita BPM, además de entender el porqué de la elección de estas.

También se decidió todo lo que se quería conseguir y se planteó lo que se iba a hacer en las iteraciones, pero sin ser de forma definitiva porque durante el desarrollo podían surgir nuevas ideas. Se realizó el diseño y la especificación del proyecto, lo bueno, es que el diseño estaba hecho porque se partía de unas herramientas utilizadas anteriormente. Una vez hecho esto nos pusimos de lleno con la primera iteración. Esta primera parte, no tenía ningún tipo de dependencia, ya que era con lo que comenzamos.

En esta etapa del proyecto se siguieron las indicaciones marcadas en la asignatura de GEP, estas servían para hacer una gestión del proyecto y constan de varias entregas y una presentación del trabajo realizado. En esta tarea se define el alcance del proyecto y sus objetivos, en qué contexto se encuentra, su planificación temporal y el presupuesto. La estimación del tiempo empleado es de unas 75h.

3.2.2. Desarrollo

Una vez se tuvo todos los objetivos marcados, el diseño y especificación hechos, se empezó a realizar el desarrollo mediante la primera iteración, esto empieza a la vez que se hace la fase de GEP. Esta fase de en medio trata de desarrollar la aplicación en sí, cada mes, como se ha comentado en el capítulo 2 en la parte de seguimiento, se realiza una reunión con los integrantes de la OMS para enseñarles el avance en esta y den feedback para la siguiente iteración.

Al final de cada iteración, se marcan una serie de objetivos que debemos cumplir para poder acabar la aplicación con tiempo. Mientras esto se va desarrollando, el autor del proyecto y su

director harán pequeñas reuniones entre semana para poder ver como va todo y hacer los cambios que sean necesarios.

Como incluye tanto el “front-end” como el “back-end” el desarrollo de las dos cosas se hacen a la vez según se necesite, eso si, tal y como se ha explicado antes es necesario que el “back-end” sea consistente y que no falle. En esta parte se utilizan con gran énfasis las herramientas ya explicadas.

Al final de cada iteración se realiza una fase de test y errores para poder solucionar cualquier problema que pueda aparecer y así no arrastrarlo hasta el final. Cada una depende de la anterior, no podrá empezar hasta que la otra haya acabado. Eso sí, la primera empieza a la vez que se hace GEP debido a que se viene de un desarrollo que lleva haciéndose unos meses, por lo tanto la primera iteración de este TFG empieza mientras se realiza la fase de GEP. Ninguna iteración empieza hasta que no se marcan los principales objetivos de estas justo al principio e inicio de GEP. Esta es la parte más costosa del desarrollo y por lo tanto, la que más horas ocupa.

Se decidió que se iban a realizar 4 iteraciones debido a que desde febrero hasta junio se iban a hacer 4 reuniones con la OMS, por lo tanto esto nos daba a pie a poder programar 4 Sprints y mostrarles lo hecho en estas reuniones.

3.2.3. Fase final

Por último se hace una revisión completa de todo lo hecho por si se encuentra algún error que aún no se haya visto.

Una vez acabado todo el desarrollo la parte final esta dedicada a acabar toda la documentación que falte, como esto no es dependiente de ninguna otra cosa se intenta hacer a la vez que el desarrollo de la app y todo lo que no se haya acabado.

Una vez hecho esto, se preparará la memoria final con toda la documentación ya realizada y revisada y se hará la presentación final para defender todo el trabajo hecho durante estos últimos meses. Esto último si que dependerá completamente de haber acabado las iteraciones.

Resumen de las tareas

Por lo tanto, y tal y como veremos más adelante en el diagrama de Gantt, se puede determinar en que orden se realizan todas las tareas y cuales dependen de otras.

Primero de todo se realiza un estudio y práctica de las herramientas que se van a usar, una vez hecho esto comienza la fase de GEP, esta depende de lo anterior, de mientras, se empieza a la vez pero un poco más tarde la primera iteración a realizar, por lo tanto, la parte de GEP no hace que la fase del desarrollo dependa de ella, eso sí, todas las iteraciones son dependientes de la anterior realizada. Una vez acabada la última iteración, se pasará a realizar la fase final del proyecto, esta depende de todo lo anterior, se acabará la documentación, se corregirán últimos problemas y se preparará la exposición.

3.2.4. Tiempo Estimado

Tarea	Duración
Creación del entorno de trabajo y pruebas	30
GEP	75
Iteración 1: Creación de los métodos de Registro	80
Iteración 2: Creación del proceso para pedir medicinas	80
Iteración 3: Nuevos procesos y cambios en el principal	80
Iteración 4: Acabar últimos detalles y añadidos menores	75
Pruebas finales	15
Finalización de la documentación	15
Documentación i preparación de la defensa	20
Tiempo Total	470

Cuadro 3.1: Estimación de horas.

3.3. Recursos

A continuación se detallan los recursos previstos, que son necesarios para el correcto desarrollo del proyecto:

Recursos humanos

- Autor del proyecto. Es el encargado de asumir todos los roles necesarios para desarrollar el proyecto correctamente, con la ayuda y supervisión del director. El tiempo medio de dedicación es de 28 horas semanales.

Recursos hardware

- Ordenador de sobremesa personalizado y creado a medida con Windows 10 y todas las herramientas necesarias.
- Ordenador Portátil ASUS GL522VW con Windows 10 y todas las herramientas necesarias.

Recursos software

- Google Drive: Gestor de documentos donde se almacenan las diferentes copias de los documentos usados durante todo el proyecto para acceder remotamente.
- Microsoft Office 2018 y Overleaf para Latex: Se utilizan para generar la documentación.
- Adobe Reader: Lector de pdf's.
- Redmine: Herramienta on-line que se usa para planificar el proyecto y las tareas a realizar en los sprints.
- BonitaStudioCommunity de BonitaSoft: El software que se utiliza para crear la aplicación, en ella hacemos tanto la parte de front-end como de back-end.
- TortoiseSVN: Herramienta que utilizamos para compartir y guardar archivos esenciales del proyecto, es parecido a un Google Drive, pero está enfocado más a proyectos de software. (Redmine, BonitaStudioCommunity y TortoiseSVN son las 3 herramientas principales que se explican anteriormente en el Capítulo 2 en el apartado de herramientas.)
- WebStorm y SublimeText 3: Para realizar del registro de usuarios desde fuera mediante JavaScript.

3.4. Valoración de alternativas y plan de acción

Para poder gestionar las alternativas y el plan de acción en el caso de una desviación importante en el calendario previsto se toma como referencia el final de cada una de las tareas anteriormente descritas.

Las desviaciones inferiores a 3 días (en la fecha prevista de finalización de una tarea) no tienen ninguna repercusión en la planificación, simplemente se ampliará el plazo de las siguientes tareas este número de días. Esto se puede hacer puesto que se ha dejado un margen de 2 o 3 semanas para la entrega final. Ahora bien, en el caso de acumular varios retrasos que sumen un total de 7 días, se tiene que hacer una replanificación para ajustar el calendario e incrementar el tiempo de dedicación en las tareas pendientes, para poder finalizar el proyecto en el plazo indicado.

Como se ha indicado, las reuniones que se realizan son muy positivas para poder ver si hemos realizado todo lo que se había propuesto y según lo que se haya hecho trazar el plan a realizar y establecer las posibles desviaciones que puedan ocurrir y saber como actuar ante ellas para no tener ningún tipo de imprevisto. Si las iteraciones se acaban mucho antes del tiempo definido, se hace una nueva iteración para añadir funcionalidades al prototipo final.

Las posibles desviaciones que puedan ocurrir afectan principalmente a los recursos de ámbito humano debido a una mala planificación, cambios en los requerimientos por parte de la OMS y que las reuniones de los sprints no están fijadas en una fecha concreta, hecho que incrementa el número de horas de dedicación semanal en los periodos finales del desarrollo, puesto que es cuando se acumulan más los posibles retrasos. Una solución es contratar a más gente para resolver inconvenientes que puedan salir, pero esto no es algo realmente necesario porque es una proyecto hecho para que lo pueda hacer una sola persona. Así que la solución es realizar más horas y lo que no se llegue a conseguir, intentar hacer un prototipo por lo menos para poder enseñarlo al final y tener una idea de lo que falta.

Partiendo del tiempo estimado y teniendo en cuenta la duración del proyecto, se han calculado unas 20 semanas aproximadamente. Teniendo en cuenta que el TFG + GEP equivale a 18 ECTS, y un ECTS son 25 horas salen unas 450 horas de trabajo, bastante aproximado al cálculo previsto del proyecto y perfectamente viable en cualquier caso, todo se tiene que poder acabar.

3.5. Diagrama de Gantt

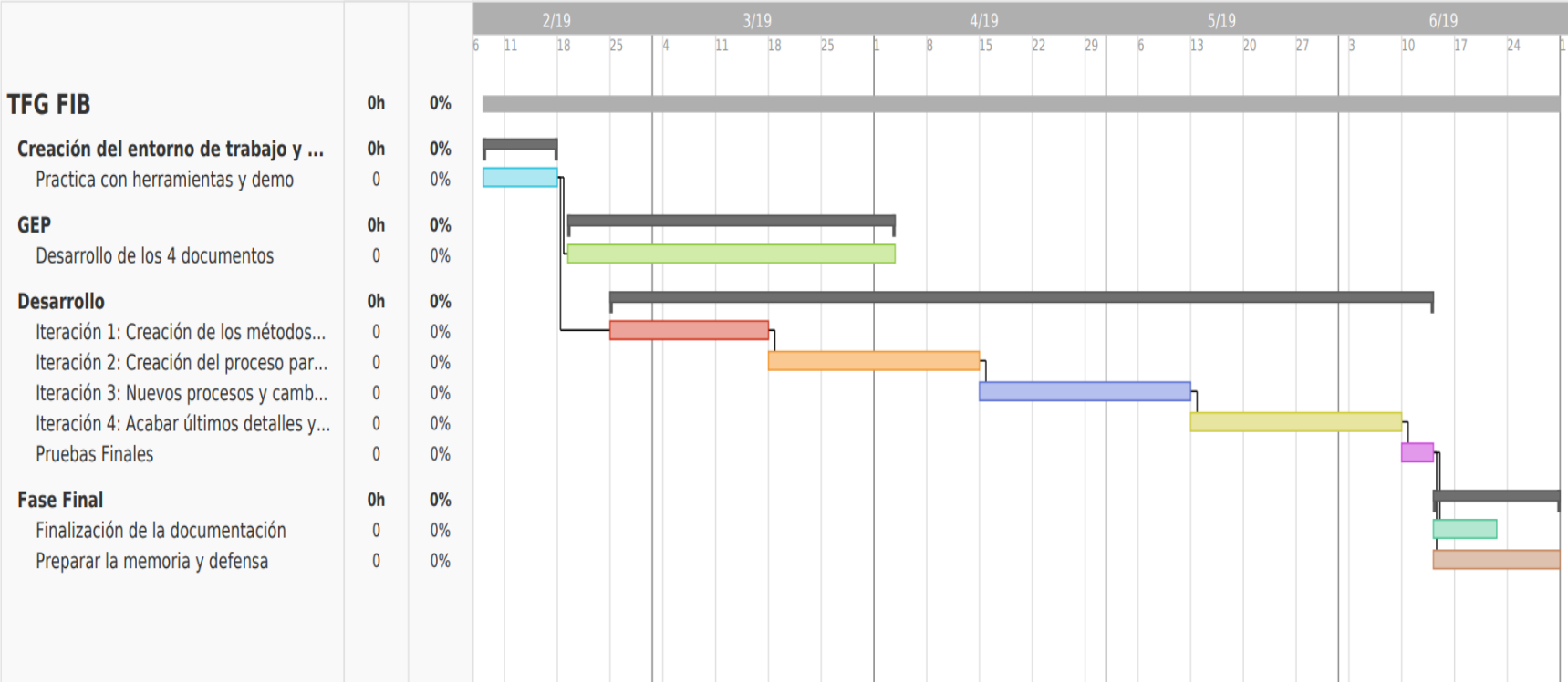


Figura 3.1: Diagrama de Gantt.

Capítulo 4

Gestión Económica

4.1. Identificación y estimación de los costes

Aunque el proyecto es parte de uno mucho más grande, se tiene solo en cuenta la parte realizada por el autor del proyecto, por lo tanto, la estimación de costes derivados de la parte de recursos humanos viene dada directamente del trabajo realizado por este. Aun así se tienen en cuenta los diferentes roles que desarrolla, por lo tanto, habrá diferentes sueldos por hora. Para realizar la estimación de costes también se valoran los recursos utilizados de hardware y de software. Para estimar todas estas variables se tendrá en cuenta el apartado anterior (que incluía el diagrama de Gantt [3.5](#)). Mediante todo esto, podremos determinar si es viable o no.

4.1.1. Recursos humanos

Cómo se ha dicho anteriormente, este proyecto lo ha realizado una sola persona, por lo tanto, esta realizará todos los roles necesarios en un proyecto software tales como el jefe de proyecto, el analista, el arquitecto de software, el programador y el tester. Para poder estimar el presupuesto necesario en recursos humanos, asignaremos un precio por hora diferente en función de las horas destinadas a cada uno de los roles, el sueldo medio lo hemos sacado de un estudio de "PagePersonnel".

A continuación, pasamos a calcular el coste humano del proyecto para cada fase y rol, a partir de la asignación de tiempo calculada en la planificación:

Rol	Precio por hora
Jefe de Proyecto	26
Analista	18
Arquitecto	18
Programador	10

Cuadro 4.1: Estimación de horas.

Fase	Tiempo de dedicación (horas)				Coste
	Jefe de Proyecto	Analista	Arquitecto	Programador	
Creación entorno y pruebas			10	20	780 €
GEP	50	20	5		1750 €
It 1: Creación del Registro	15	5	20	40	1240 €
It 2: Creación del “Request Process”	15	5	20	40	1240 €
It 3: Nuevos procesos y cambios	15	5	20	40	1240 €
It 4: Acabar últimos detalles	20		15	40	1190 €
Pruebas finales				15	150 €
Finalizar documentación	2	4	4	5	350 €
Defensa	2		8	10	520 €

Cuadro 4.2: Estimación de costes por fase y rol.

Esta posiblemente es una de las tablas más importantes, como podemos observar, en esta están todas las fases del proyecto explicadas en el capítulo 3, y que además se mostraban en el diagrama de Gantt 3.5, como se puede ver, primero tenemos la parte inicial de estudio y prueba de las herramientas a utilizar, después la fase de Gep que como recordamos, esta se hace más o menos a la vez que la primera iteración del proyecto. Después como se puede observar se indican las 4 iteraciones y a continuación la fase final. En cada una se ve que indica las horas en las que participa cada uno de los roles del proyecto y como queda todo dividido con los gastos para cada parte, de esta forma podemos hacernos una idea de la cantidad de dinero que puede llegar a costar todo el proyecto.

A continuación se puede ver el coste asignado a cada uno de los roles:

Rol	Horas previstas	Precio por hora	Coste estimado
Jefe de Proyecto	119	26	3094 €
Analista	39	18	702 €
Arquitecto	102	18	1836 €
Programador	210	10	2100 €
Total	470	-	7732 €

Cuadro 4.3: Estimación de costes por rol.

4.1.2. Recursos software/hardware

En este apartado se detallan los costes de los diferentes recursos software y hardware a utilizar en el transcurso del desarrollo del proyecto. Del conjunto de recursos necesarios, solo se tienen en consideración aquellos que tienen un coste para su uso. Para establecer un coste de amortización por hora de uso se coge como referencia que 1 año de vida útil son 249 días hábiles a un ritmo de trabajo de 8 horas diarias. Coste de amortización = Precio / (4 años * (249 días * 8 horas/día)) Por otro lado, la estimación en de uso en horas para cada recurso es la siguiente:

Producto	Precio	Unidades	Vida útil	Amortización	Coste estimado
Ordenador de sobremesa con W10	1700 €	1	4	0,2133534 €/h	362,70078 €
Portátil con W10	800 €	1	4	0,1004016 €/h	80,32128 €
Microsoft Office 2018	115€	1	4	0,014432730 €/h	1,65975 €
Total	2615 €	1	-	-	444,6817 €

Cuadro 4.4: Estimación de costes por software y hardware.

Algunos de los recursos software son utilizados a partir de las licencias gratuitas que dispone la Universidad, pero para aproximarse al coste de un proyecto real se tienen en cuenta las licencias como si fueran no gratuitas. Por otro lado no se ha contado aquel software que se ha utilizado como habíamos indicado anteriormente pero que es gratuito y no comporta ningún gasto económico.

4.1.3. Otros

Otros gastos a tener en cuenta son:

- Consumo energético del ordenador de sobremesa y portátil.

- Servicio de conexión a Internet con 100 MB de bajada y de subida.
- Desplazamiento con transporte público a Barcelona, para reuniones y otras a la facultad.

Descripción	Precio	Cantidad	Coste
Consumo energético	0,15 €/kWh	Consumo diario de 1,5 kWh para 135 días	30,37 €
Internet	70 €/Mes	4 meses y medio	315 €
Transporte	105€	T-Jove de 1 zona	105 €
Total	-	-	450,37 €

Cuadro 4.5: Otros costes.

4.1.4. Imprevistos y plan de contingencia

En todo proyecto pueden surgir una serie de imprevistos que podrían afectar en su planificación y a su presupuesto. En este caso, los principales imprevistos detectados son:

- Avería de los ordenadores principales donde se desarrolla el proyecto. En este caso, el coste previsto de reparación se asumirá que es de 60€ con una probabilidad del 15 % que ocurra.
- Posible pérdida de datos. La probabilidad es remotamente baja, puesto que se utilizan servicios externos al "Google Drive" para almacenar todas las copias.
- Otros problemas con servicios externos (conexión eléctrica o Internet), hecho que provocaría no poder desarrollar el proyecto durante las horas de avería. La probabilidad estimada que ocurra será del 5 % y en el supuesto de que esto pase se asumirá que se pierde un día de trabajo. Por lo tanto, con una estimación de 4 horas de trabajo diario y con una media de coste de los diferentes roles de 17€/hora, el coste del imprevisto será de 68€.
- Como medida de contingencia se establece un margen del 7 % sobre el coste total.

4.1.5. Costes totales

Recurso	Coste estimado
Recursos humanos	7732 €
Recursos no humanos	444,6817 €
Otros	450,37 €
Imprevistos	128 €
Coste Parcial	8755,0517 €
Contingencias	+ 7%
Coste Total	9367,905319 €

Cuadro 4.6: Costes totales.

4.2. Control de Gestión

En cuanto al control de los recursos hardware y software, los gastos incluidos en los apartados anteriores son precios fijos puesto que es el precio de venta al público establecido por la marca de cada uno de los productos. Por lo tanto, en este aspecto no tenemos ninguna sorpresa, a no ser que se tenga que utilizar un producto no previsto en la planificación. En cuanto al coste total del producto, contando amortización en horas, no se puede calcularlo hasta el final del proyecto puesto que estos se utilizan durando casi todo el desarrollo.

Por otro lado, en el caso de los recursos humanos, al final de cada una de las tareas se contabilizan las horas reales dedicadas para ver las posibles desviaciones respecto la planificación temporal y así ajustar, si hace falta, el presupuesto del proyecto.

Para poder hacer un cálculo de las desviaciones, se utilizan los siguientes cálculos:

- En la realización de una subtarea en precio = $(\text{coste estimado} - \text{coste real}) * \text{consumo horas reales}$.
- En el precio de un recurso = $(\text{coste estimado} - \text{coste real}) * \text{consumo real}$.
- En la realización de una subtarea en consumo = $(\text{consumo horas estimado} - \text{consumo horas reales}) * \text{coste estimado}$.

Desvíos en importes totales:

- En la realización de subtareas = total coste estimado subtarea - total coste real subtarea.
- En recursos = total coste estimado recurso - total coste real recurso.
- En los costes fijos = total coste fijo presupuestado – total coste fijo real.

Capítulo 5

Sostenibilidad y compromiso social

5.1. Autoevaluación

Después de realizar la encuesta, me he dado cuenta de todos los conocimientos sobre la temática de sostenibilidad y compromiso social, he reflexionado sobre todos los conocimientos que ha aprendido durante la realización de distintos cursos que me ayudaran a encontrar soluciones para que mi proyecto sea sostenible y cumpla un papel en el desarrollo social.

Considero que durante mis estudios no he aprendido suficiente sobre sostenibilidad y compromiso social para tener en cuenta todos los elementos que pueden afectarme durante la realización de este proyecto, y he notado con esta encuesta que hay varios conceptos que no conocía bien, por esto creo que durante el curso se debería hacer mucho más hincapié en todo esto en las diferentes asignaturas que se van realizando. Mi idea es que esto se aplicase de una forma más específica en todas aquellas asignaturas en las que se realice un proyecto.

Por otro lado, hay algunas cosas que ya conocía por haber visto algo de todo esto en asignaturas como PES, GPS y ER, en las que realizábamos proyectos y se hacía hincapié en algunos de estos aspectos en ellas pero no de la forma en que me gustaría como he comentado antes.

A continuación reflexionaremos sobre diferentes ámbitos dentro de la sostenibilidad y el compromiso social.

5.2. Económica

En cuanto a la parte económica del proyecto, se puede concluir que existe una evaluación de costes, tanto de recursos humanos como de materiales. Además, se han tenido en cuenta las contingencias y los posibles imprevistos. En este aspecto, se incluye el coste de reparaciones de los productos, que se han previsto en los imprevistos.

Respecto a la viabilidad del proyecto, si este tuviera que estar en un mercado competitivo, no se ha contemplado. Esto se debe a que este proyecto es de ámbito académico y no es para competir contra otras empresas y, por lo tanto, no se puede valorar.

En cuanto a la posibilidad de desarrollar el proyecto con menor tiempo, todo depende de los recursos que se inviertan, puesto que si en vez de estar desarrollado por una sola persona se realiza en equipo, el tiempo se reduce significativamente. Esta reducción de tiempo, pero, implica un aumento considerable en los costes.

5.3. Social

La situación actual del país no es muy favorable puesto que desde hace muchos años estamos inmersos en una crisis muy importante. Los niveles de paro son muy elevados y esto provoca que el nivel de vida de aquellas personas que se encuentran en esta situación no sea nada favorable. En cambio, en el sector de la informática parece ser que funciona muy bien. El nivel de paro en el sector es bajo y la demanda de profesionales es más elevada que la oferta, hecho que provoca una situación bastante estable dentro de lo que cabe.

Por lo tanto este proyecto es beneficioso para diversas partes, donde una de ellas es el autor, que después de casi 4 años de carrera, puede aplicar todo lo aprendido en un proyecto real y de esta forma ganar experiencia de cara al mundo laboral.

Este también ayudará a muchas personas de países sobre todo tercermundistas a recibir medicamentos para la cura de sus enfermedades mucho antes gracias a la mejora en la organización de pedidos que facilitará esta aplicación.

5.4. Ambiental

Los recursos necesarios para el desarrollo del proyecto son los diferentes aparatos electrónicos y la energía que estos consumen, aparte de la conexión a Internet. Como consecuencia de utilizar estos materiales, se está consumiendo unos recursos que, al ser electrónicos, comportan una serie de efectos perjudiciales por el medio ambiente (producción de electricidad o el mantenimiento de servidores de Internet). Ahora bien, este impacto es extremadamente pequeño, por lo tanto, podemos considerar que es despreciable.

Al ser uno proyecto software no se requiere ningún coste de fabricación y tampoco nos tenemos que preocupar por el reciclaje. Este proyecto no pretende disminuir el impacto ambiental existente en el planeta, no es su finalidad, es por eso que el proyecto no perjudicará el aspecto ambiental.

Para futuros proyectos en los que si que tenga un gran impacto se intentará reducir a toda costa.

5.5. Matriz de sostenibilidad

Para realizar este análisis nos hemos basado en la matriz de sostenibilidad y sus preguntas que nos muestran en el “Mòdul 2.6 EL INFORME DE SOSTENIBILIDAD”, de la asignatura de GEP, la matriz es la siguiente:

	PPP	Vida Útil	Riesgos
Ambiental	Consumo de diseño	Huella ecológica	Ambientales
Económico	Factura	Plan de viabilidad	Económicos
Social	Impacto personal	Impacto social	Sociales

Figura1. Matriz de Sostenibilidad del TFG

Figura 5.1: Matriz de sostenibilidad. “Mòdul 2.6 EL INFORME DE SOSTENIBILIDAD”

Capítulo 6

Bonita BPM

6.1. ¿Qué es Bonita BPM?

Como se ha comentando a lo largo de toda esta tesis, ha llegado el momento de dejar claro que es Bonita BPM, el programa que se va a utilizar. Este es un paquete de ofimática para la Gestión de procesos de negocio (BPM) y realización de Flujos de trabajo, creado en 2001. Con el, se han podido crear los procesos que forman parte de la aplicación que se ha creado para el pedido de medicamentos.

Bonita BPM ofrece dos ediciones: la Community Edition y la Subscription Edition. La primera se proporciona de forma gratuita y se eligió su versión 7.8.1 para el desarrollo de este proyecto. La de suscripción incluye características más avanzadas tales como soporte profesional, personalización completa de look & feel y herramientas de monitorización, sin embargo, esta versión tiene un coste *significativo*¹.

Esta fue elegida debido a la baja curva de aprendizaje, de esta forma su aspecto visual y su funcionamiento permitiría a personas que no trabajan en el ámbito de la informática poder hacer un uso de ella mediante un aprendizaje no muy exhaustivo.

Bonita BPM está compuesto por el Bonita BPM Studio y la Bonita BPM Platform. Esta última está compuesta de 3 partes más, el Bonita Engine, el Bonita User Experience (portal web) y el UI

¹No hay un precio definido. El usuario que quiera comprarla necesitará hablar con Bonita y según el proyecto que se quiera realizar y lo grande que sea su precio variará

Designer. A continuación se explicará todo esto de forma más detallada.

6.2. Bonita BPM Studio

Bonita BPM Studio es el programa que permite crear los procesos basados en Business Process Management, mediante unas herramientas que da el propio programa. Además de esto, proporciona lo necesario para crear aplicaciones, el Business Data Model (BDM), las páginas y los formularios.

Este consiste en una pizarra blanca (ver Figura 6.1) en la cual se pueden diseñar los procesos con las herramientas que son proporcionadas desde el menú de desarrollo que ofrece, todo esto desde la aplicación de escritorio. Aparte, como se mencionó antes, existe el UI Designer, que es una herramienta basada en web.

En esta pizarra blanca se pueden pasar los procesos a diagramas incluyendo toda la lógica necesaria para que estos funcionen correctamente y que cumplan los requisitos establecidos en el proyecto. Para diseñar estos diagramas se utilizan unos elementos que proporciona el programa, estos siguen unos estándares de BPMN². En Bonita BPM Studio estos elementos están clasificados en diversos grupos, estos son: Swimlanes, Gateways, Flow, Tasks, Activities, Start Events, Internal Events, End Events y Text Annotations.

Con respecto al menú de desarrollo (ver Figura 6.2), con este se pueden hacer varias cosas, entre ellas establecer los conectores, la BDM, acceder al UI Designer, etc. El BDM o Business Data Model, como antes se ha explicado, compone la base de datos, esta es un conjunto de varios BDMs que son convertidos en variables llamadas "variables de negocio" que son definidas para cada proceso por separado según sea necesario. Con respecto a los conectores, es algo muy a tener en cuenta, ya que estos, divididos en varias categorías (ver Figura 6.3), proporcionan varias herramientas para poder adaptar los procesos a los requerimientos y necesidades que puedan surgir en el desarrollo de estos y aumentar las propias capacidades que da el Bonita BPM Studio.

²Business Process Model and Notation

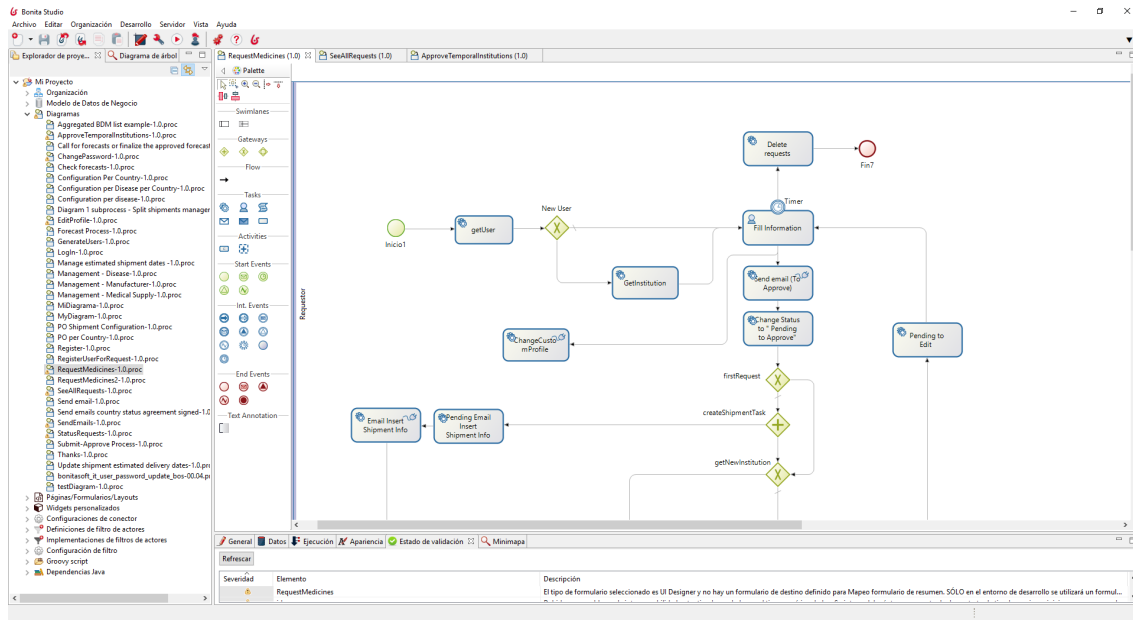


Figura 6.1: Captura de Bonita Studio, con la pizarra y los elementos con los que diseñamos los procesos.

Una vez se ejecute el proceso, los datos según sean utilizados se guardarán automáticamente de forma persistente.

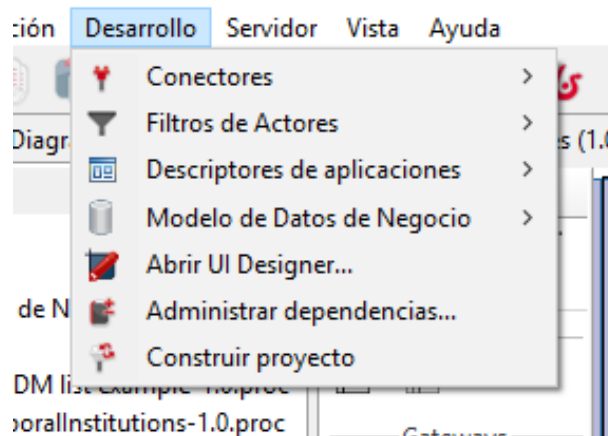


Figura 6.2: Captura de Bonita Studio, menú de desarrollo.

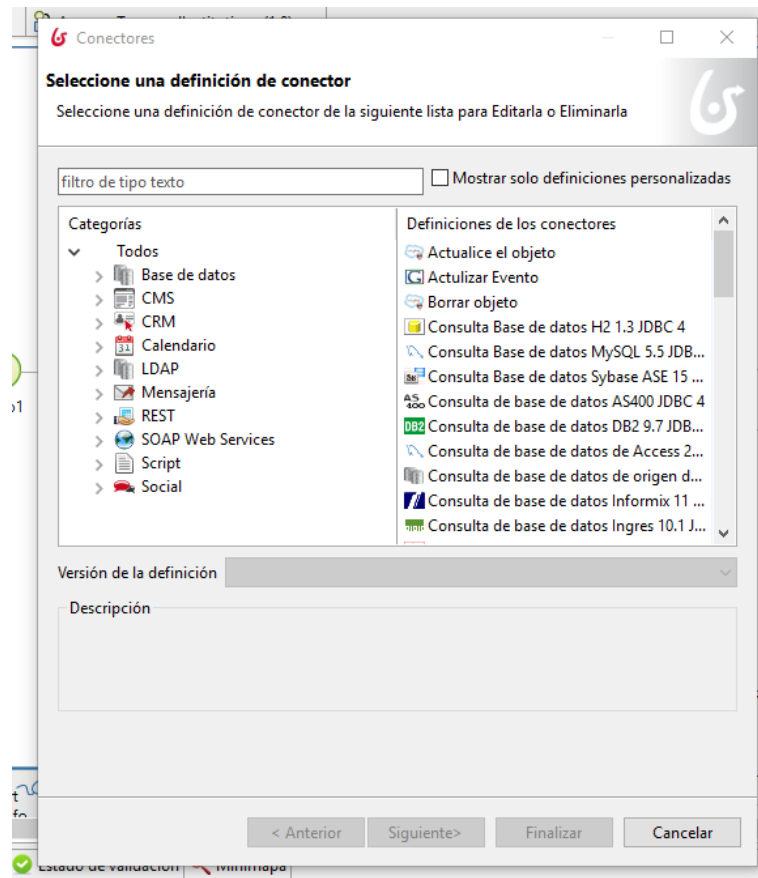


Figura 6.3: Captura de Bonita Studio, categorías de los conectores.

También accedemos al UI Designer, donde se realiza todo el desarrollo del front-end del programa mediante las páginas, formularios y widgets que proporciona la herramienta. Más adelante se hablará de ellos de forma más profunda.

Y por último dentro de este Bonita Studio, se pueden definir que usuarios tendrán permisos para poder ejecutar estos procesos y quien será el encargado de ejecutar cada tarea.

6.3. Bonita BPM Platform

Una vez hablado del Studio se ahondará más en este programa, y para ello pasaremos a hablar de su Platform, esta está dentro del propio Studio y está compuesta del UI Designer, Bonita BPM Portal, Tomcat Server, H2 Database y Bonita BPM Engine.

6.3.1. UI Designer

UI Designer es una aplicación basada en web (ver Figura 6.4), desde la cual se pueden diseñar los diversos recursos, como las páginas, formularios y layouts mediante las herramientas que nos proporcionará esta misma, es decir, toda la parte del front-end de la aplicación.

Este UI Designer permite la creación de varios tipos de elementos, estos son los recursos mencionados justo antes además de los widgets personalizados.

- **Páginas:** Estas son los elementos que componen las aplicaciones, pueden ser por ejemplo, las páginas de bienvenida, de arranque de un proceso, entre otras.
- **Formularios:** Existen 3 tipos de formularios, estos son: el formulario de instanciación del proceso, el formulario de tareas humanas y los formularios de resumen o Overview. Los primeros instanciarán el proceso al cual están asociados una vez se envíen. Los segundos, estarán asignados a una o más tareas humanas que serán completadas una vez se envíe este formulario.
- **Layouts:** En estos se puede diseñar el look & feel de todas las páginas de la aplicación.
- **Widgets personalizados:** para crear estos tres últimos elementos, UI Designer nos facilita una serie de elementos llamados widgets que se podrán utilizar y distribuir por la página, además, si es necesario, se podrán crear los nuestros propios, estos serán los widgets personalizados.

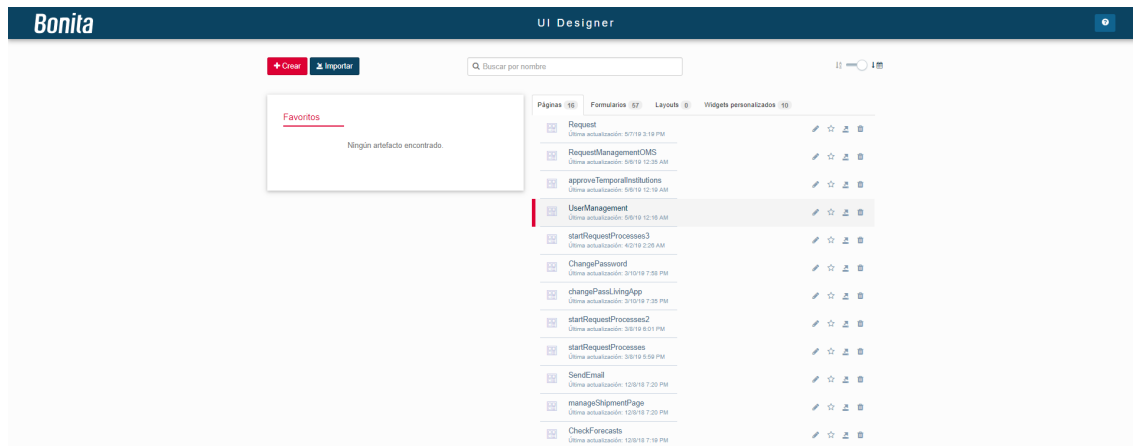


Figura 6.4: Página principal del UI Designer

Como se ha citado, una aplicación esta formada por varias páginas y cada una de estas páginas, aunque no tienen porque ser todas, está de una forma u otra ligada a un formulario, el layout de esta será compartido por todos en la aplicación. Además las páginas y formularios estarán formadas de widgets y/o widgets personalizados. Por último especificar que los formularios podrán pertenecer a cero o varias tareas dentro de los procesos y estas a su vez pueden contener o no formularios. (Podemos ver esto representado en la Figura 6.5)

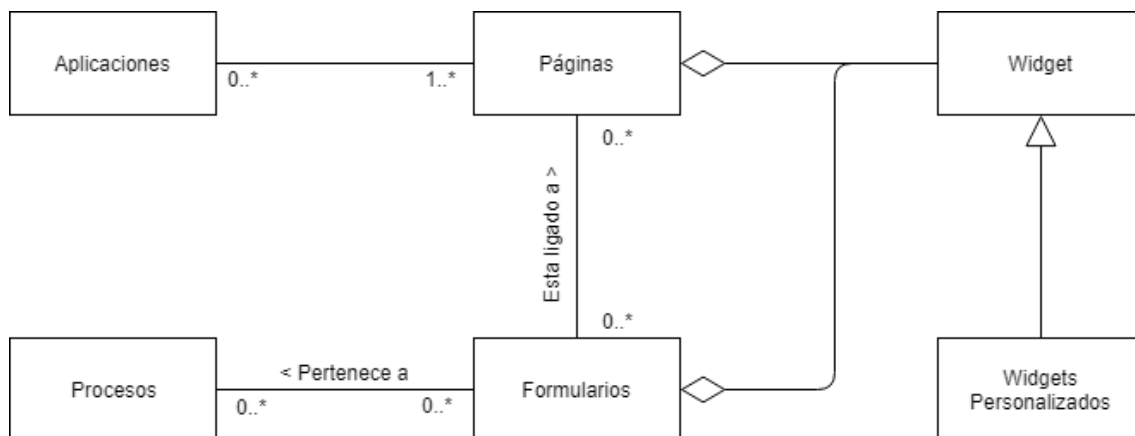


Figura 6.5: Diagrama de las Aplicaciones, Páginas y Formularios

Para editar estos diversos elementos se dispone de 2 tipos de editores diferentes:

- **Editor de Formularios, Páginas y Layouts:** Este está compuesto por un menú de widgets, otro donde estarán las variables y los assets, otro donde se podrá modificar las distintas propiedades de los widgets y una pizarra blanca donde se colocarán los widgets y se diseñará el aspecto visual y como estará colocado todo (ver Figura 6.6).

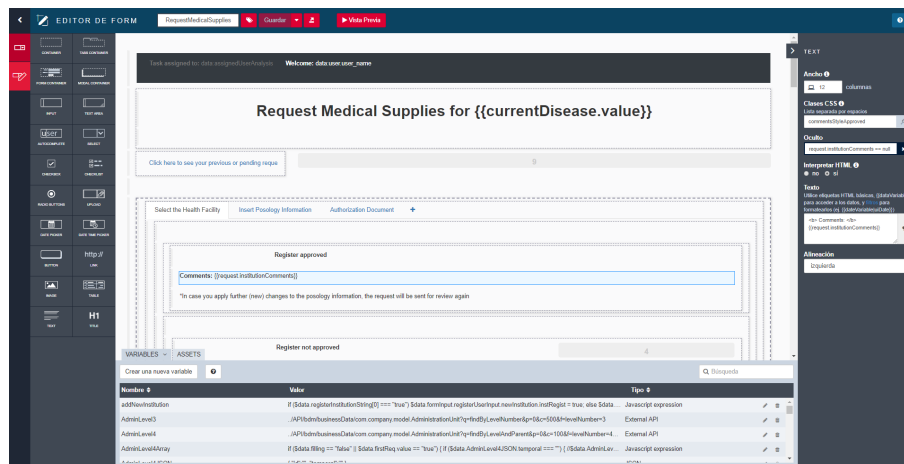


Figura 6.6: UI Designer, Editor de Páginas, Formularios y Layouts

- **Custom widgets editor:** Este otra está compuesto por varias partes, dos de ellas en la parte izquierda servirán para el código javascript y el código html y las otras tres partes para los assets, angular modules y las propiedades. Una vez acabado su diseño se podrá utilizar en cualquier página, formulario o layout (Figura 6.7).

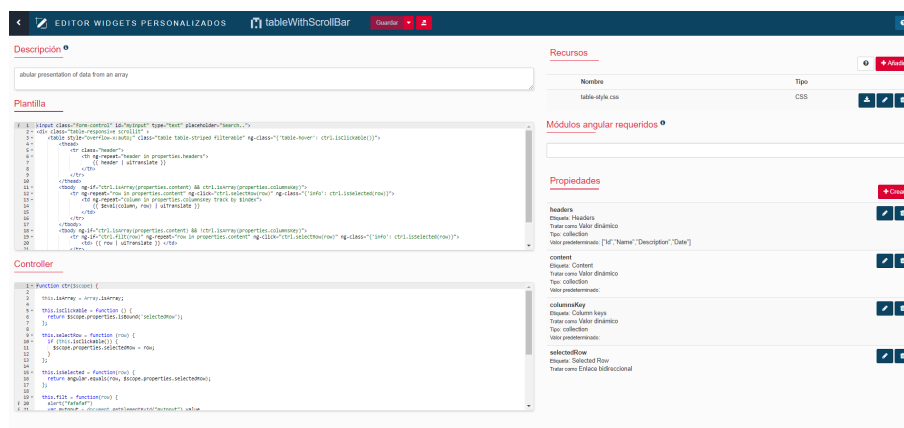


Figura 6.7: UI Designer, Editor de widgets

6.3.2. Bonita User Experience

Con el nombre de Bonita BPM Portal es como se conoce realmente este Bonita User Experience, esta es una herramienta basada en web con la que los usuarios finales interactuarán con los procesos creados en BPM Studio. También será utilizada por los administradores técnicos que se encarguen de instalar todo (Tenant Administrator), por ello, según el usuario con el que se haga “log in”, tiene unas funcionalidades u otras. Estas funcionalidades son dadas al usuario según su perfil, estos pueden ser Usuarios y/o Administradores.

Perfil de Usuario

Los perfiles denominados como Usuarios, tienen tres funcionalidades principales. Pueden realizar tareas que les han sido asignadas, visualizar el estado de los diferentes casos existentes y empezar procesos. Además, pueden acceder a las diferentes aplicaciones en las cuales tengan permisos para poder acceder siempre y cuando tengan el enlace para utilizarlas. (ver Figura 6.8)

Como se ha dicho antes, estos usuarios han de tener los permisos necesarios en los propios procesos para poder interactuar con ellos.

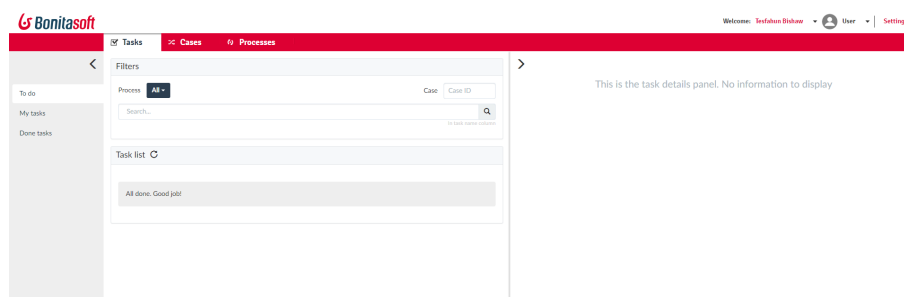


Figura 6.8: Captura de Bonita User Experience de un perfil de Usuario.

Perfil de Administrador

A diferencia del perfil de Usuario, un Administrador puede manejar todo lo relacionado con los BPMs, a parte de hacer lo mismo que los Usuarios, puede no solo ejecutar los procesos, sino modificarlos y añadir o borrar otros.

Además puede modificar toda la organización establecida, es decir, los usuarios registra-

dos dentro de Bonita BPM, los Grupos, los Roles, los Perfiles existentes y por último puede importar/exportar el esquema de la organización.

Por último puede añadir y modificar recursos, estos serán las páginas, formularios, layouts, temas y las extensiones de la REST API y crear las aplicaciones, estas están formadas de diversos recursos que se deben añadir de forma previa, estas pueden estar restringidas a uno de los Perfiles. (Figura 6.9)

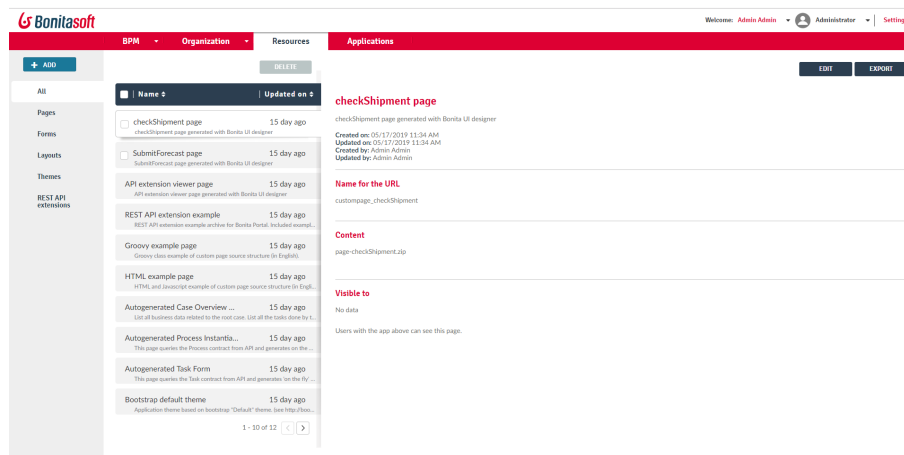


Figura 6.9: Captura de Bonita User Experience de un perfil de Administrador.

Tenant Administrator

Este usuario, es uno que no este asignado a ningún tipo de perfil, este usuario solo tendrá un “username” y un “password”, que estará configurado desde el fichero de configuración de Bonita. Este puede modificar la organización como el Perfil de Administrador, puede modificar y añadir o borrar recursos y como funcionalidad extra, cambiar el BDM (Business Data Model) es decir, las bases de datos, y pausar y poner en ejecución los servicios BPM. (ver Figura 6.10)

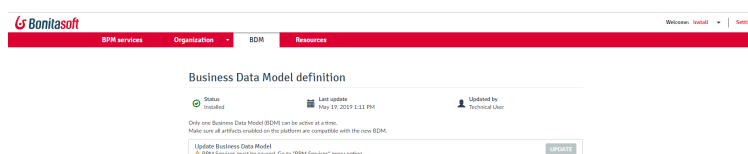


Figura 6.10: Captura de Bonita User Experience de un Tenant Administrator.

6.3.3. Bonita Engine

Bonita Engine es el procesador de todo Bonita, es el que ejecuta los procesos una vez estos se inician, maneja las tareas, el logging, etc. Está formado por APIs, Servicios BPM y Servicios genéricos (ver Figura 6.11). Este es provisto con 3 archivos .jar: bonita-common, bonita-server, bonita-client.

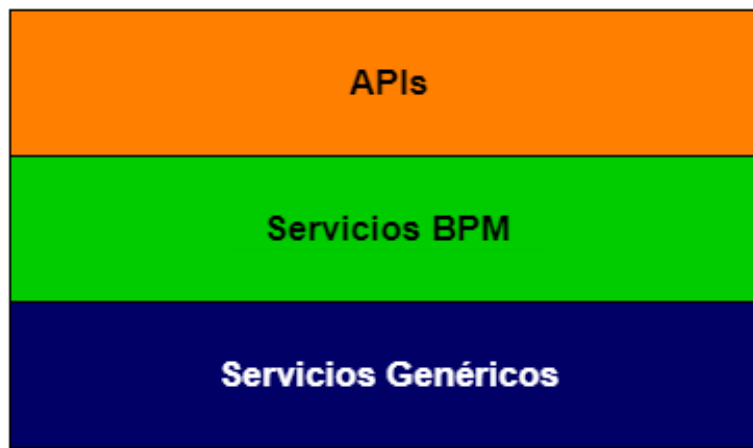


Figura 6.11: Arquitectura de Bonita Engine

APIs

Según su función están clasificadas en una categoría, estas son: Identity, Organization, Process, Login, Monitoring, Log, Platform command, Document, Theme, y Tenant Management.

Servicio BPM

Este se encarga de manejar y organizar todas las actividades que se ejecutan cuando se hace “build” y se pone en ejecución un proceso BPM. Los servicios incluyen: actor mapping, category, connector, expression resolver, login, platform login, operation, parameter, process comment, process definition, engine, process, instance, supervisor mapping, y user filter.

Servicios genéricos

Aquí se incluyen todos aquellos que no son BPM pero que son necesarios para un correcto funcionamiento de los procesos en ejecución, además estos no tienen dependencias con los

servicios BPM. Estos son: Archive, Authentication, Querieable logs, Tenant cache, Plataform cache, ClassLoader, Platform command, Connector executor, Data, DataInstance, Dependency, Platform dependency, Document, Expression...

6.4. Otros Requerimientos de Bonita

Para poder poner todo en marcha, se necesita un servidor y una base de datos. En el propio Bonita viene incluido tanto el servidor en Tomcat como la base de datos en H2.

Tomcat

“Apache Tomcat Java EE application server” está incluido en el paquete de tomcat dentro de la descarga de Bonita BPM.

Base de Datos H2

En ella se guardan todos los Objetos de Negocio (o Business Object) y los metadatos de Bonita, esta base de datos debe ser cambiada por un RDBMS³ cuando ya no se ejecute desde el propio Bonita Studio y se quiera instalar el servidor con todos los procesos.

³Relational database management system

Capítulo 7

Desarrollo del proyecto

A continuación se hablará de todo el desarrollo del proyecto y se explicará detalladamente cada parte de este mismo.

7.1. ¿Qué vamos a adaptar?

Primero de todo y lo más importante es dejar claro que se va a adaptar, así que se va a explicar como es el proceso de pedir medicamentos hoy en día, es decir, sin aplicaciones, que es básicamente lo que se quiere adaptar. Este proyecto en un principio estaba pensado para la enfermedad Chagas, pero más adelante se decidió que se aplicaría para más enfermedades, en concreto para las enfermedades tropicales desatendidas NTDs (Neglected Tropical Diseases), el objetivo de esto es intentar erradicarlas. La erradicación significa eliminar la presencia de esta enfermedad de cualquier lugar del planeta.

Por lo tanto, es necesario poder servir estos medicamentos a los países que los necesiten y los pidan en el menor tiempo posible, a continuación explicaremos el proceso.

Cuando una institución ve que necesita pedir una serie de medicamentos que la propia Organización mundial de la salud (OMS) se encarga de distribuir, lo primero que hará es rellenar un documento que esta última da para poder hacer estos pedidos. En él se tiene que poner todo lo necesario para ello, es decir, primero de todo, hay que informar de que institución somos, dando toda la información necesaria como la de la persona que lo pide. Después, para pedir

estos medicamentos, se debe dar la información del paciente al cual van estos dirigidos y que medicamento se quieren pedir para este, además, esta información del paciente y medicamentos tendrán que cumplir una serie de requisitos, ha de ser coherente según lo que se pide, por ejemplo, no se pueden pedir unos medicamentos que no tienen nada que ver ni pueden hacer nada con la enfermedad que sufre dicha persona, ni se podrá pedir más medicamentos de los necesarios, entre otras cosas. Para acabar, se han de añadir una serie de documentos con referencia al medicamento pedido, serán necesarias unas autorizaciones por parte del cliente para poder enviárselos a su país, es decir, que puedan entrar por las aduanas de este mismo.

Una vez rellenado todo esto, se envía mediante email a la OMS, y desde allí, unas personas encargadas, decidirán si los pedidos se aceptan o se "rechazan" (el pedido no se acepta pero tampoco se descarta, más adelante se explicará), en el primer caso, si la petición es aceptada, otro encargado se responsabilizará de ir a buscar los medicamentos que se piden, una vez ya los tienen, estos se empaquetan junto al documento que el cliente ha enviado y se llevan a la sección de envíos donde otro se ocupará de dárselos a la empresa de transporte que será la encargada de hacer que estos lleguen a su destino. Una vez lleguen, el cliente deberá confirmar la recepción de estos, si hay algún problema podrá ponerse en contacto con la OMS, vía teléfono o email.

En el segundo caso, si esta petición no es aceptada por la OMS, la persona que ha tomado esta decisión enviará un mail al cliente en cuestión explicándole detalladamente porque no se ha aceptado esta petición. Una vez este lo recibe, podrá realizar los cambios necesarios para volver a realizar esta petición e intentar conseguir los medicamentos que solicita.

Si por un casual hay un problema grave, el cliente puede llamar directamente a la OMS para pedir los medicamentos que sean necesarios.

Como se puede ver, esto no es eficiente, y si hay muchos pedidos puede llegar a provocar muchas confusiones y retrasos en estos debido a que todo se hace de forma manual, por lo tanto en este punto vemos la necesidad de tratar de pasar esto a un proceso digital en el cual hacer todo mediante una aplicación y automatizar todo lo máximo posible.

7.2. Actores y roles del proyecto

El concepto de actor se define como el conjunto de usuarios que:

- desempeñan un determinado papel, o
- pertenecen al mismo grupo, o
- tienen un cierto papel en el mismo grupo.

Esta última definición es la que se ha aplicado para hacer el mapeo de los actores en todas las tareas humanas del sistema. Este mapeo determina el conjunto de usuarios que se asignarán a la tarea. El nombre del actor se define, en la mayoría de los casos, por el nombre del rol debido a que el grupo mapeado al actor puede ser dinámico dependiendo de la instancia del proceso.

Los siguientes actores participan en el flujo del proceso para pedir medicinas:

- **Requestor** Es el que se encarga de realizar la petición, serán los usuarios finales que usarán esta aplicación para pedir los medicamentos.
- **Programme manager** Es el actor que se encargará de hacer la valoración de la petición solicitada por el usuario y además, hará la gestión de las instituciones temporales que más adelante se explicará de forma más profunda, hay por lo menos uno por enfermedad.
- **Technical officer.** Es el actor a cargo de el envío de los suministros médicos. Será el que se encargue de coger los medicamentos y enviarlos al cliente, también pondrá todo los datos referentes al envío, hay al menos uno por enfermedad.

A continuación podemos ver una representación del proceso con los distintos roles en él (Figura 7.1).

cliente en concreto para que active su cuenta. Este email contiene una dirección a otra página web donde se le activará la cuenta y ya podrá acceder a usar la aplicación, esta activación, se realiza como el registro, con una llamada API.

Además como añadido a esto, en la página web del registro, hay un enlace el cual nos envía a otra en la que se puede hacer reiniciar de la contraseña y la nueva se envía al email del usuario de forma automática por el sistema.

En la siguiente imagen, podemos ver una representación de lo explicado (Figura 7.2) y en las figuras 7.3 y 7.4 las páginas en cuestión.

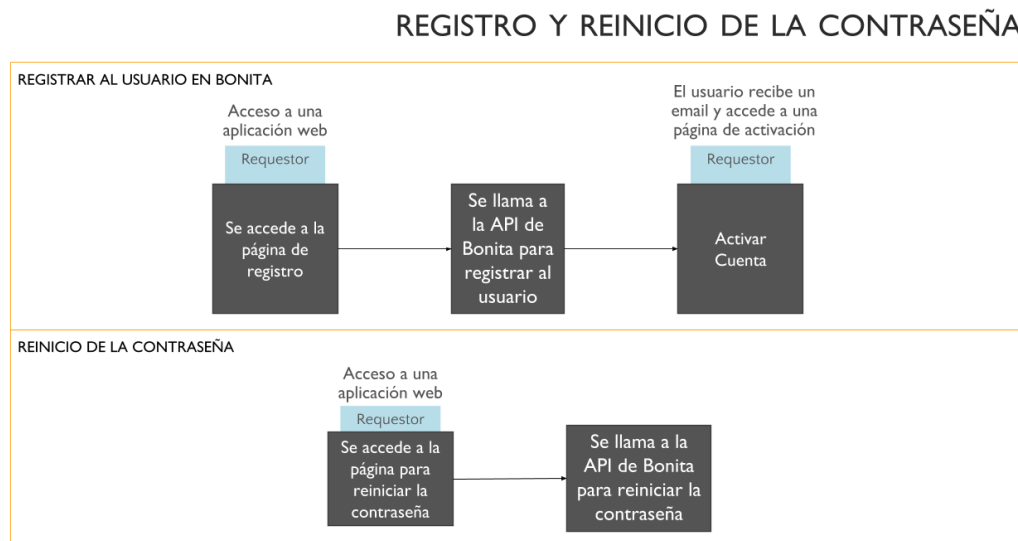


Figura 7.2: Flujo del registro y reinicio de contraseña

WELCOME TO THE REGISTER PAGE FOR REQUEST PROCESS APP

User Registration Form

Name *

Surname *

Telephone *

E-mail (IT WILL BE THE USERNAME FOR LOG IN) *

Password *

Repeat Password *

99897

Type the above number:

Register

[Forgot password ?](#)

Figura 7.3: Página de registro

FORGOTEN PASSWORD

Reset your Password

Please, write your email:

Reset Password

[Back to login Page](#)

Figura 7.4: Página para reiniciar la contraseña

7.4. Objetos de negocio "Business Objects"

Los Objetos de negocio que componen la BDM (explicada en la Sección 6.2) son la principal fuente de datos para todas las instancias de proceso del sistema. Es por eso que para definir un proceso primero necesitamos crear los Objetos de negocio que participarán en él. Durante la ejecución del proceso estos se actualizarán, eliminarán o, al menos, se comprobarán. Todos los objetos de negocio creados en el sistema y las relaciones entre ellos se especifican en la Figura 7.5

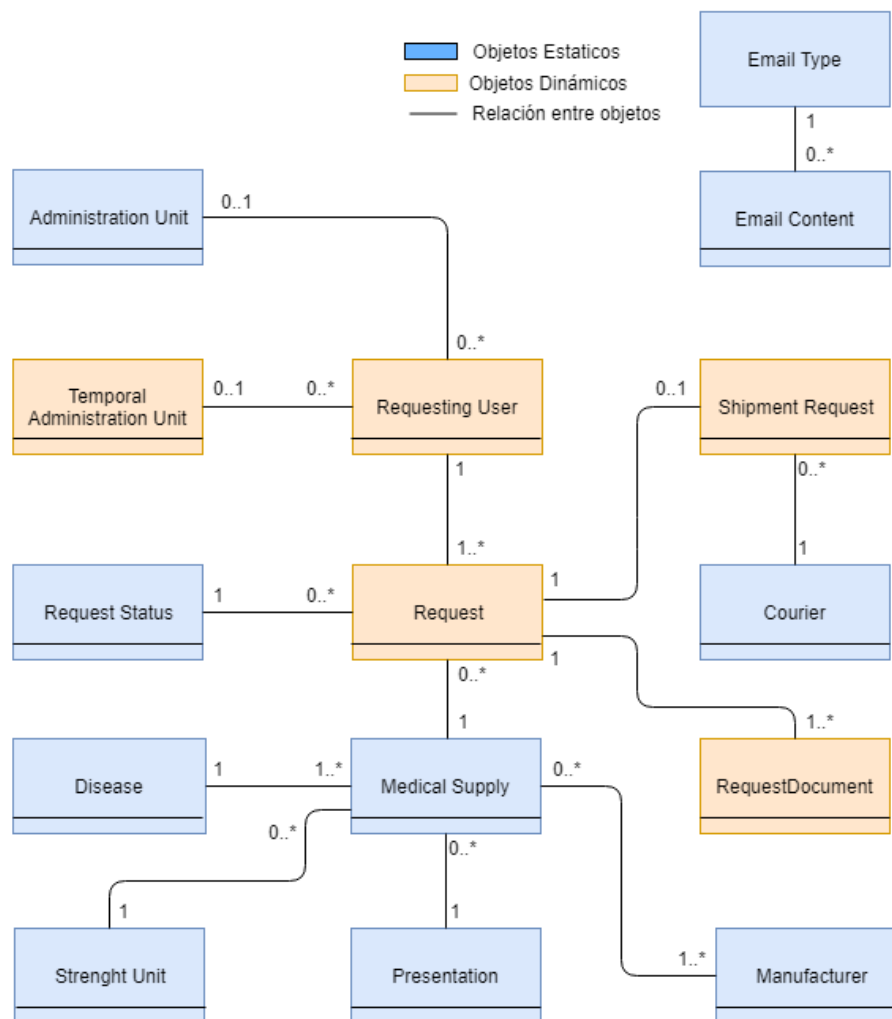


Figura 7.5: Diagrama de los objetos de negocio

- Los BO estáticos contienen objetos de negocio que se insertan una vez y rara vez se actualizan / eliminan. Estos son, por ejemplo, el "Administration Unit", que contienen las unidades administrativas de los países involucrados (extraído de una aplicación externa

utilizada dentro de OMS), y también objetos de negocio como “Medical Supply“, que almacenan información sobre los suministros médicos disponibles en la OMS, y los “Manufacturer“, que contiene información sobre los productores de los suministros médicos entre otros.

- Los BO dinámicos son los objetos de negocio o los datos que se manipulan constantemente a través de los procesos. Contienen información, por ejemplo, todos los datos necesarios para pedir los medicamentos y diferentes decisiones tomadas por los diferentes actores.

Como se puede ver en la Figura 7.5, tenemos un Objeto de Negocio llamado Administration Unit y otro llamado Temporal Administration Unit, esto es así porque como más adelante explicaremos, los usuarios pueden crear instituciones si este no está en la base de datos, entonces, esta nueva se guarda en la de los temporales debido a que por consistencia no se puede guardar en la “oficial“, y cuando esta se guarda en una base de datos de la OMS, el Programme Manager actualiza la temporal y la añade donde están las otras.

7.5. Procesos diseñados

Ahora lo explicado en la sección 7.1 ha de ser traducido a un conjunto de procesos formales que deben modelarse a través de BPMN e insertarse en Bonita BPM.

Estos procesos han sido los siguiente:

- Procesos principales para la petición: “Request Process“, “Send Email“, “Manage Temporal Institutions“
- Procesos secundarios: “See Requests“, “Edit Profile“ y “Change Password“

Se ha decidido dividirlos de esta forma para explicarlos mucho mejor. En los procesos principales tenemos aquellos que son totalmente esenciales para el correcto funcionamiento del pedido de medicamentos, sin estos sería imposible o difícil su puesta en marcha. En cambio en los secundarios, tenemos unos procesos que complementan a estos primeros y facilitan a los usuarios el uso de esta aplicación.

Seguidamente se explicarán uno por uno comenzando por los principales.

7.6. Procesos principales

7.6.1. Petición de medicamentos "Reques Process"

Una vez explicado todo lo necesario para adaptar el proceso principal, a continuación lo explicaremos detalladamente, en la Figura 7.7, podemos ver el proceso en cuestión en su globalidad.

Una vez se ha completado el registro el usuario puede acceder a la aplicación, cuyo enlace a esta, el sistema se lo ha pasado con anterioridad por email al registrarse. Al arrancar un proceso de pedido de medicamentos primero de todo, seleccionará para que enfermedad va a pedir medicamentos, estas estarán guardadas en la base de datos y se mostraran en una lista al usuario (ver Figura 7.6).

Al darle al botón de "Submit" se envían los datos necesarios para el arranque de este proceso y se nos redirecciona a un formulario que está compuesto de tres partes que a continuación vamos a detallar.

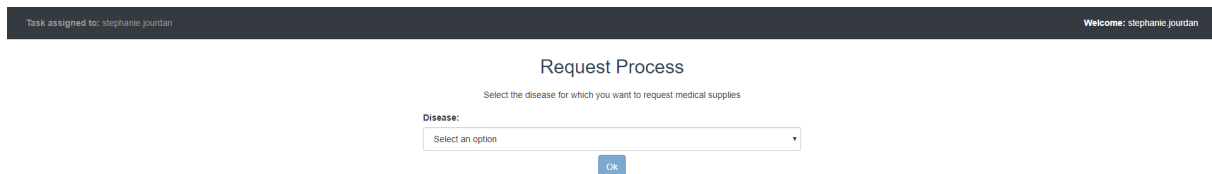


Figura 7.6: Formulario para iniciar el proceso "Request Process"

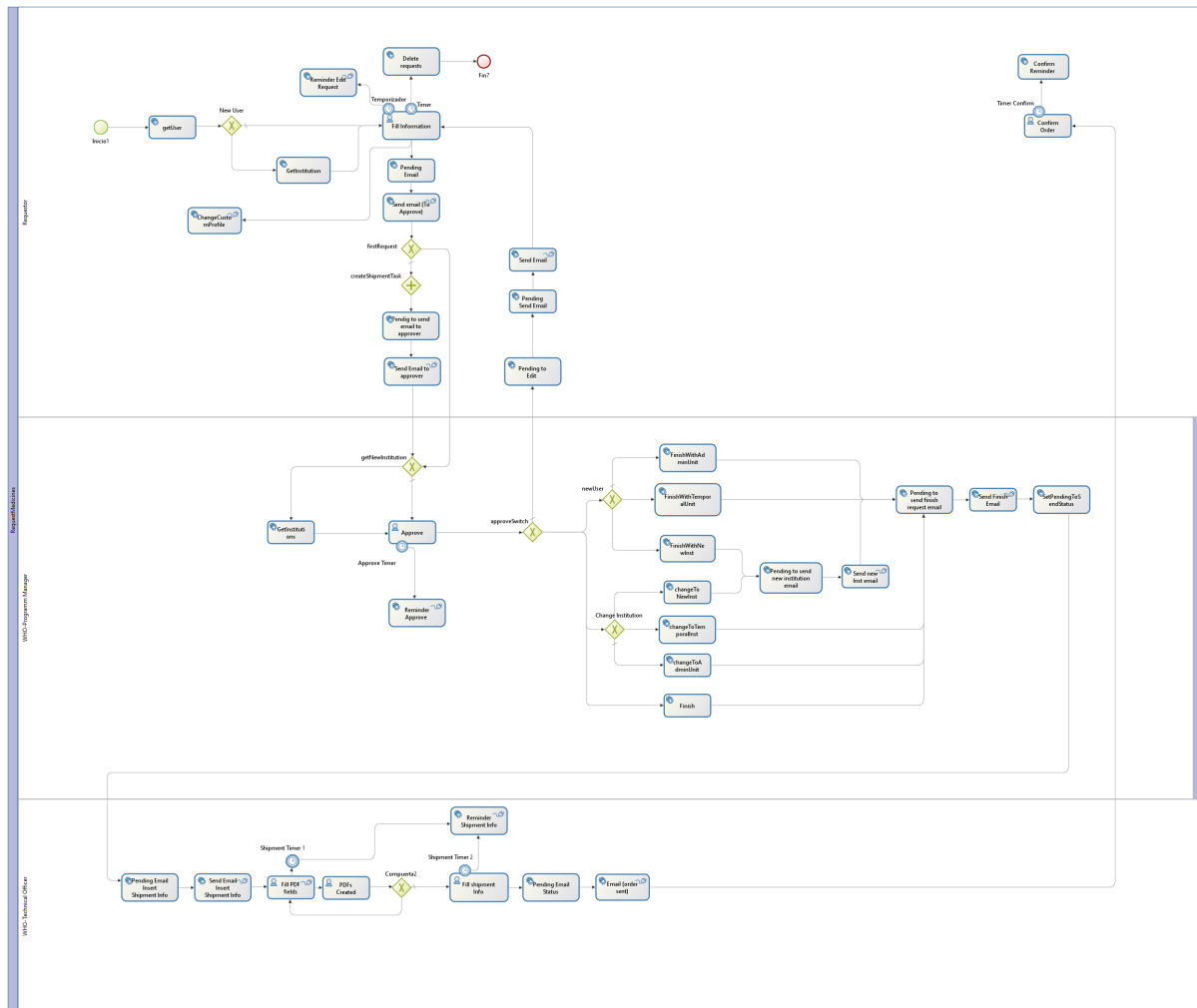


Figura 7.7: Diagrama del proceso "Request Process"

Rellenar la información del pedido

Como acabamos de comentar, en este punto se debe rellenar un formulario compuesto de tres partes, estas son las siguientes:

- **Institución médica:** Primero de todo, el sistema comprueba si el usuario que acaba de acceder es nuevo o va a realizar su primer pedido, es decir, si ya está registrado con una institución en la base de datos, donde se hace esta relación, o no.

En todo caso, si no es nuevo y está realizando un pedido más, esta información no es necesaria ponerla, en el propio formulario podrá ver que ya está en una institución, se la mostrará y no le obligará a hacer nada en este apartado, pero si quiere, podrá cambiar esta

institución.

En cambio, si es nuevo o quiere cambiarla, si que tendrá que rellenar una serie de campos (ver Figura 7.8). Primero de todo, se le muestra una serie de listas donde tiene que ir seleccionado las unidades administrativas de donde está su institución hasta alcanzarla y seleccionarla en la lista, todo esto se hace en forma de árbol, se accede desde el país en el que se encuentra hasta la institución, pasando por cada unidad administrativa del propio lugar.

Si por un casual no encuentra su institución en la última unidad administrativa que se le presenta, puede solicitar la creación de la institución debido a que esta no está en la bases de datos. Para ello bastará con colocarse en la unidad administrativa en la que se supone que debe estar y darle al botón de "Add new Health Facility", una vez hecho esto, verá que se le muestra un nuevo formulario que debe rellenar para poder registrarla. No todos los campos a rellenar de este formulario son obligatorios, así que puede ser que haya algunos que no pueda rellenar por no saber su valor.

Una vez hecho esto, y asegurándose de que todo está correcto, pasará a la siguiente ventana del formulario completo de la página, es decir, la Posología, donde indicará la información del paciente y el medicamento a pedir.

- **Posología:** En este apartado, como su nombre indica, debe poner todo el tema de la posología, es decir, el medicamento que quiere pedir, sus unidades, y toda la información del paciente, como se puede ver en la Figura 7.9, casi todo es obligatorio menos la información adicional que se puede poner.

Los medicamentos a escoger están guardados en la base de datos y estos se muestran al usuario en una lista, depende del medicamento escogido se le recomendará al usuario una cantidad que puede llegar a pedir, si pide más de la cuenta o mucho menos, debe explicar el porqué.

- **Documentos de autorización:** En este último apartado se deben subir todos los documen-

tos (ver Figura 7.10) de autorización que sean necesarios para poder hacer el pedido, estos, como se ha explicado antes, permitirán su entrada en el país donde se solicitan. Solo es obligatorio poner uno, pero si fuese necesario se pueden añadir más.

Una vez añadido todo esto, se podrá enviar la petición para que sea evaluada, antes de completar el envío se le muestra un resumen de toda la información colocada, como un ticket se tratase (ver Figura 7.11)

En un principio estos 3 formularios estaban divididos en 3 tareas diferentes, esto era así porque al inicio se pensó que sería mejor que el usuario fuese uno por uno poniendo la información necesaria sin saltarse nada, y una vez enviado esto, el programme manager, como se verá a continuación, también realizaba la aprobación en tres partes diferentes, esto, al enseñárselo a la OMS en una de las reuniones, pidieron que se hiciese en un único formulario ya que facilitaba todo tanto al usuario como a ellos porque visualmente se entendía mucho mejor de esta forma.

Task assigned to: stephanie.jordan

Welcome: stephanie.jordan

Request Medical Supplies for Chagas

Click here to see your previous or pending requests

Select the Health Facility

Insert Posology Information

Authorization Document

Current Health Facility

Name (of the health facility)	Short name (of the health facility)	Country	Street/Place / Number / Zip Code / Municipality
Hospital del Mar (Consorci Mar Parc de Salut de Barcelona)	Hospital del Mar	Spain	Passeig Marítim de la Barceloneta / 25 / 08003 / Barcelona

Select Health Facility

Country*

First Administrative Division

Second Administrative Division

Badakhshan

Baharak

*If you can not find your Health Facility under the selected administrative unit, please add a new one by clicking the Add new Health Facility button.

Add new Health Facility

Add a new Health Facility to Baharak

Name (of the health facility)*

Short name (of the health facility)*

Street/Place*

Number*

Zip code*

Municipality*

Country*

Code (national code of the health facility)

URL (Website address of the health facility)

Latitude

Longitude

Cancel

Cancel

Additional Shipping Information

Name of the Department

Name of the Unit

Floor

unit

Any additional information/notes useful for the shipment

Figura 7.8: Formulario de la institución médica

Task assigned to: stephanie.jordan

Welcome: stephanie.jordan

Request Medical Supplies for Chagas

Click here to see your previous or pending requests

Select the Health FacilityInsert Posology InformationAuthorization Document

Selected Disease: Chagas

Selected Health Facility:

Patient's epidemiological end clinic information

Age*	Weight*
<input type="text" value="0"/>	<input type="text" value="0"/>
Requested medicine *	Number of packages*
<input type="text" value="Select an option"/>	<input type="text" value="0"/>
Probable transmission way*	Probable place of infection*
<input type="text"/>	<input type="text"/>
Place of birth*	Phase*
<input type="text"/>	<input type="text"/>
Additional information	
<input type="text"/>	

Submit Request

Figura 7.9: Formulario de la Posología

Task assigned to: stephanie.jordan

Welcome: stephanie.jordan

Request Medical Supplies for Chagas

Click here to see your previous or pending requests

Select the Health FacilityInsert Posology InformationAuthorization Document

Selected Disease: Chagas

Selected Health Facility:

Documents to send

Authorization Document 1 (.pdf extension) *

Upload an additional Document

Submit Request

Figura 7.10: Formulario para los documentos

Request Medical Supplies for Yaws

Are you sure you want to submit the following request?

New Health Facility

Health Facility Name	Health Facility Short Name	Municipality	Street/Place	Number	Zip Code	Country
Hospital del Mar (Consorci Mar Parc de Salut de Barcelona)	Hospital del Mar	Barcelona	Passeig Marítim de la Barceloneta	25	08003	Spain

Posology information

Age	Weight	Medical Supply	Num Pack	Transmission way	Infection Place	Place Birth	Phase	Add Info
50	75	Az 95 g Pills	2	transmission Way	place of Infection	place of Birth	phase	add Info

Authorization documents

Document 1: form.pdf

Shipping Information

Street	Number	Zip Code	Municipality	Country
Passeig Marítim de la Barceloneta	25	08003	Barcelona	Spain

Figura 7.11: Ticket con el resumen de la solicitud

Aprobar petición y reenvío de esta

Una vez enviada la petición, esta llega al Programme Manager y este se encarga de aceptarla o por otro lado rechazarla para que el cliente la vuelva a mandar con los cambios demandados.

Se deben aprobar las tres partes del formulario por separado, es decir, cada apartado tiene un aprobado o no y además se pueden enviar comentarios de cada uno en específico (ver Figura 7.12).

Si el cliente no es un nuevo usuario ni quiere cambiar su institución, esta ya estará aprobada y solo será necesario aprobar la posología y los documentos de autorización. Como pasa en el formulario de la petición, antes de enviar las decisiones tomadas y los comentarios, sale un resumen de confirmación de todo lo puesto (ver Figura 7.13).

Una vez se ha enviado esta resolución, se comprobará si ha sido aprobada o no, para que esté aprobada todo tiene que estar aprobado, si hay algo rechazado, se rechaza la petición en general y se le "devuelve la pelota" al cliente, este recibe un mail automático del sistema donde se le comunica que su petición ha sido rechazada y un enlace donde poder acceder al formulario de la petición para poder revisarla y cambiar lo que sea necesario. Se le mostrará tanto si esa parte ha sido aprobada o no además de los comentarios respectivos (ver Figura 7.14 y Figura 7.15).

Task assigned to: None - Go to the portal to take the task

Welcome: stephanie.jordan

Approve Request for Chagas

Approve Health Unit | Approve Posology Information | Approve Authorization Documents

Disease: Chagas | Health Facility: Hospital del Mar (Consorci Mar Parc de Salut de Barcelona)

Patient's epidemiological and clinic information

Age	Weight	Medical Supply	Number of Packages	Probable Transmission Way	Probable Place of Infection	Place of Birth	Phase
50	75	Az 95 g Pills	2	transmission Way	place of infection	place of Birth	phase

Additional information

add info

Approve *

Approve

Comments

Submit

Figura 7.12: Aprobación del primer apartado de la solicitud

Approve Request for Chagas

Are you sure you want to submit the following decisions and comments?

Health Facility decision: Discard

Comments: La solicitud de la institución es errónea

Posology decision: Approve

Authorization documents decision: Approve


Yes Cancel

Figura 7.13: Ticket con el resumen de las decisiones tomadas

Si este realiza algunos cambios en partes ya aprobadas, estas deben ser aprobadas de nuevo, una vez el cliente haya cambiado lo necesario volverá a mandar la petición como había hecho antes y esta se volverá a poner en revisión.

Si la petición aprobada contiene una nueva institución registrada por el cliente, será necesario añadirla a nuestra base de datos junto a todas las otras. Primero esta estará en una base de datos temporal distinta a la general como habíamos comentado antes, y una vez se haya añadido por el Programme Manager, este se encargará de hacer el cambio de los usuarios unidos a la temporal por la nueva.


Como se ha estado explicando durante todo la tesis, es necesario que estos usuarios estén relacionados de una forma u otra con una institución, esto es así porque necesitamos saber en todo momento quién y desde donde se están pidiendo estos medicamentos, validando esto evitamos de forma mas eficiente los pedidos no deseados.

 Documents not approved

Comments: Los documentos son incorrectos

* Please edit your request based on the comments received and resubmit it.

Figura 7.14: Documentos rechazados

 Posology approved

Comments: La posología es correcta

*In case you apply further (new) changes to the posology information, the request will be sent for review again

Figura 7.15: Posología aprobada

Información del envío

Cuando la petición haya sido aprobada por el Programme Manager, el sistema le enviará un mail tanto al cliente como al Technical Officer, al primero se le dirá que la petición ya ha sido aprobada y al segundo que debe enviar y rellenar la información del envío de los medicamentos solicitados.

A este último se le muestran dos nuevos formularios, en el primero debe rellenar una serie de campos que después generarán un PDF que será obligatorio su inclusión en el envío (ver Figura 7.16), un vez hecho esto, se le muestra otro formulario donde puede descargar todos los PDFs necesarios (ver Figura 7.17), en la otra podrá ver el medicamento que se ha pedido y la información de envío necesaria y se le pedirá que rellene una serie de datos referentes al envío tales como el número de referencia y la empresa de envío, una vez hecho esto, cuando envíe el último formulario, significará que el pedido ha sido enviado y se le mandará un mensaje al cliente para comunicárselo y que cuando lo reciba lo confirme desde un nuevo formulario (ver Figura 7.18) que se le mostrará, accederá a este desde la aplicación para ver los pedidos o por el enlace que le hayamos dado junto al mensaje del envío.

Task assigned to: stephanie.jourdan
Welcome: stephanie.jourdan

Create PDF for dispatch

From

Name*

Div Or Unit*

Date

Enter a date (mm/dd/yyyy) Today

Telephone*

PTAEO*

Medical Supplies

Medical Supply

Az 95 g Pills

Batch No*

Expiry Date*

Enter a date (mm/yyyy) Today

Quantity

2

Quantity of*

Additional Shipping Details

Sw. Frs

Dispatched By

☐ Economy Mail
☐ Registered Matter
☐ Priority

☐ Pouch
☐ Courier Service
☐ Surface Freight

☐ Air Freight

Submit

Figura 7.16: Formulario para la creación del PDF

Task assigned to: stephanie.jourdan
Welcome: stephanie.jourdan

PDFs

Request's document

Document 1: Tesis TFG.pdf Click here to download it!

Created PDF for shipment

ShipmentDocument.pdf Click here to download it!

Submit Change PDF

Figura 7.17: Formulario para descargar los PDFs necesarios en el envío

Task assigned to: stephanie.jourdan
Welcome: stephanie.jourdan

Confirm Order

Requested Medicines

Medical Supply	Quantity	Add Info
Az 95 g Pills	0	addInfo

Shipment Information

Courier	Tracking Number	Shipped Date	Estimate Day of Delivery
SEUR	aef	2019-06-26	2019-06-26

Quantity Received

Reception Date

Reception Comments

Figura 7.18: Formulario para la confirmación de los pedidos

7.6.2. Enviar email "Send Email"

Para poder activar al usuario o reiniciar su contraseña si este lo pide, ha sido necesaria la creación de un proceso el cual será llamado en el momento que o bien se registre un usuario o este solicite el reinicio de su contraseña, el proceso en cuestión es el siguiente y como se puede ver no tiene ningún formulario es decir, una Human Task (ver Apéndice A.2).

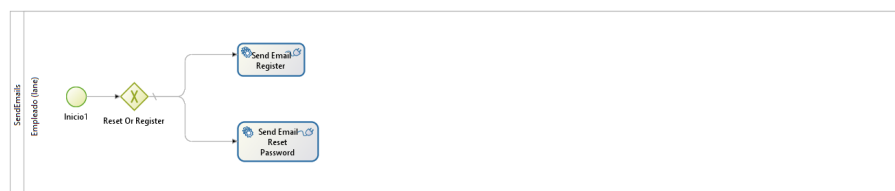


Figura 7.19: Esquema del proceso "Send Emails"

7.6.3. Gestión de las instituciones temporales "Manage Temporal Institution"

Como se ha explicado con anterioridad, cuando se aprueba una institución temporal, el actor con rol de Programme Manager, deberá, aparte de guardarla en la base de datos de las oficiales, reasignar a todos los usuarios que estaban asignados a la temporal por la nueva, esto se hará mediante este pequeño proceso (ver Figura 7.20). Se le mostrará una tabla donde estarán todas estas y deberá seleccionar su correspondiente de las oficiales mediante las listas de opciones que se le presentan, una vez hecho esto, deberá confirmar la elección y se realizará este cambio en todos los usuarios (ver Figura 7.21).

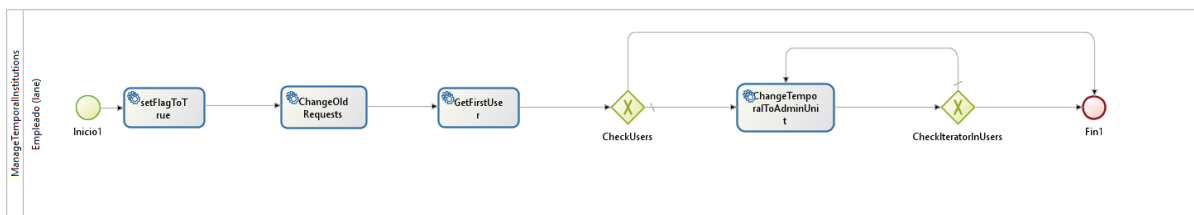


Figura 7.20: Esquema del proceso "Manage Temporal Institution"

Task assigned to:
Welcome: stephanie.jourdan

Temporal Institutions

Search...

Name (of the health facility)	Short name (of the health facility)	Country	Address	Added
Hospital del Mar (Consorci Mar Parc de Salut de Barcelona)	Hospital del Mar	Spain	Passeig Marítim de la Barceloneta / 25 / 08003 / Barcelona	false

More info

If the temporal institutions has been added to Administration Unit list, please select it for delete the temporal institution and make the needed changes.

First Level

Select an option

Submit

Figura 7.21: Formulario del proceso "Manage Temporal Institution" para reasignar la institución temporal

7.7. Procesos secundarios

A parte del los principales también hay otros que serán útiles.

7.7.1. Proceso para revisar los pedidos "See Requests"

En este proceso (ver Figura 7.22), el usuario podrá comprobar todos los pedidos que ha realizado y sus respectivos estados, además, cuando haya recibido uno de los pedidos, deberá confirmarlo rellenando un pequeño formulario al cual accederá con un botón.

Si accede un Programme Manager también tendrá un formulario similar a la del usuario normal donde podrá consultar todas las peticiones realizadas por cualquier usuario, eso si, no podrá hacer nada con ellas.

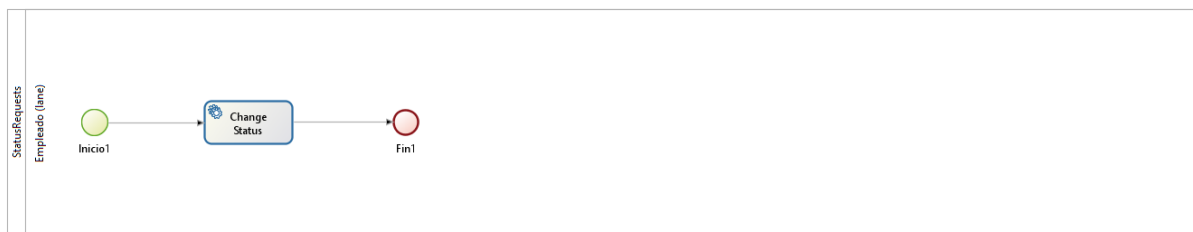


Figura 7.22: Diagrama del proceso "See Requests"

Back

Welcome: stephanie.jordan

User's requests

Search...

Id	Disease	Medical Supply	Status	Request Date
161	Yaws	Az	Pending to be send	Jun 25, 2019 10:06:45 AM
133	Chagas	Az	Order sent	Jun 24, 2019 11:44:31 PM
132	Chagas	Az	Order sent	Jun 24, 2019 10:53:26 PM
131	Yaws	Az	Pending to be send	Jun 24, 2019 10:46:53 PM
130	Yaws	Az	Pending to be send	Jun 24, 2019 10:44:28 PM
129	Chagas	Az	Pending to be send	Jun 24, 2019 10:38:04 PM
106	Chagas	Az	Pending to be send	Jun 24, 2019 5:57:58 PM
103	Chagas	Az	Order sent	Jun 24, 2019 5:26:24 PM
101	Chagas	Az	Pending to Approve	Jun 24, 2019 3:05:31 PM
100	Yaws	Az	Pending to Approve	Jun 24, 2019 11:15:12 AM
70	Yaws	Az	Pending to Approve	Jun 24, 2019 2:27:39 AM
69	Yaws	Az	Pending to Approve	Jun 24, 2019 2:22:11 AM
67	Yaws	Az	Pending to Approve	Jun 24, 2019 2:00:39 AM
65	Yaws	Az	Pending to Approve	Jun 23, 2019 9:20:18 PM
33	Chagas	Az	Pending to Approve	Jun 23, 2019 3:02:31 PM

More info

Figura 7.23: Formulario del proceso "See Requests"

7.7.2. Proceso para editar el perfil "Edit Profile"

Los usuarios registrados, una vez hayan activado su cuenta podrán editar su perfil en un pequeño proceso (ver Figura 7.24), en el, podrán cambiar diferentes datos personales como también el nombre de usuario e información adicional para el envío de medicamentos (ver Figura 7.25). Mediante una "Service Task" (ver Apéndice A) realiza unas llamadas a la API de Bonita para cambiar los datos del perfil del usuario.

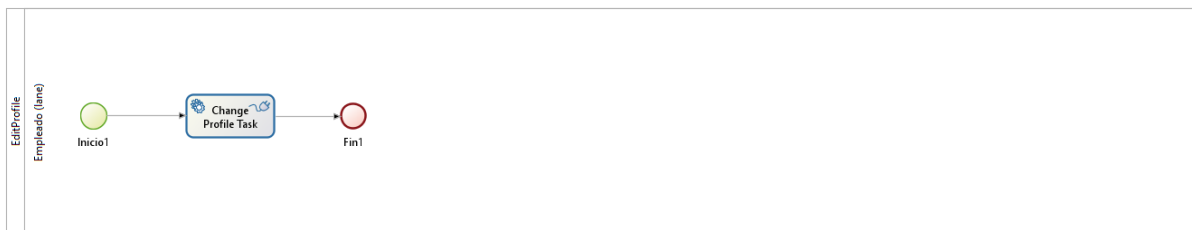


Figura 7.24: Diagrama del proceso "Edit Profile"

[Back](#) Welcome:

Update Profile

Institution name: Hospital del Mar
(Consorci Mar Parc de Salut de Barcelona)

Profile

Name

Surname

Telephone

Email

Additional Shipment Information

Name of the Department/Service

Name of the Unit

Floor

Shipment Information

Figura 7.25: Formulario del proceso "Edit Profile"

7.7.3. Proceso para cambiar la contraseña "Change Password"

Los usuarios podrán cambiar su contraseña rellenando un pequeño formulario (ver Figura 7.27) del proceso (ver Figura 7.26) en cuestión muy simple. Como se puede ver en el diagrama, tenemos una "Service Task" (ver Apéndice A) que una vez enviados los datos del formulario, realiza una llamada a la API de Bonita para modificar la contraseña.

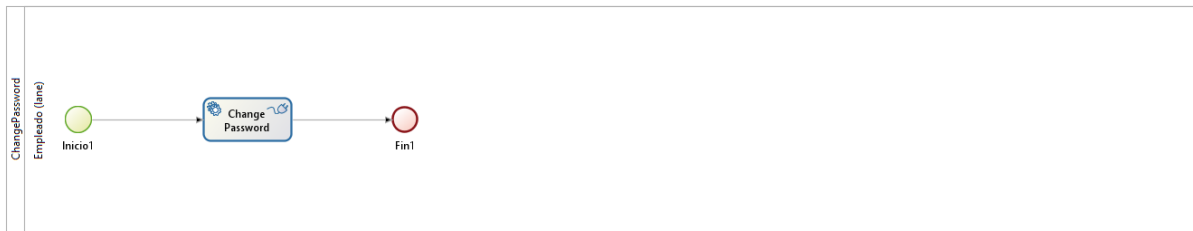


Figura 7.26: Diagrama del proceso "Change Password"

Back Welcome: stephanie jourdan

Change password

Please, insert your new password

New password

Confirm your new password

Confirm new password

Submit

Figura 7.27: Formulario del proceso "Change Password"

7.8. Aplicaciones

Como se ha explicado antes, Bonita permite crear aplicaciones que están formadas por páginas que a su vez, estas nos redirigen a formularios que inician procesos.

En este proyecto se han creado tres aplicaciones para cubrir los requisitos propuestos y que los usuarios finales puedan acceder a los procesos creados, estas son las siguientes:

7.8.1. Aplicación para la petición de los usuarios

Esta es la aplicación principal del sistema y esta dividida en dos partes a las cuales se accede mediante el menú superior, en la primera (ver Figura 7.28) está lo referente a la petición de medicamentos, es decir, hay dos enlaces a los que se puede acceder, el primero permite iniciar un proceso de petición de medicamentos y el segundo consultar las peticiones que haya realizado el usuario que haya accedido a esta.

La otra parte (ver Figura 7.29) esta pensada para la configuración del usuario en concreto. En esta hay otros dos enlaces, el primero permite editar el perfil del usuario en cuestión que haya entrado y el otro cambiar la contraseña actual del mismo.

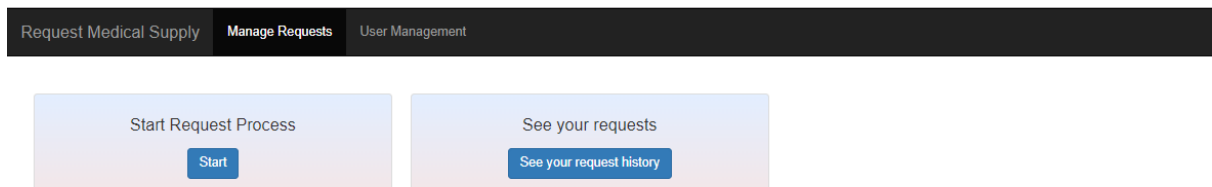


Figura 7.28: Aplicación para la petición de los usuarios "Manage Requests"

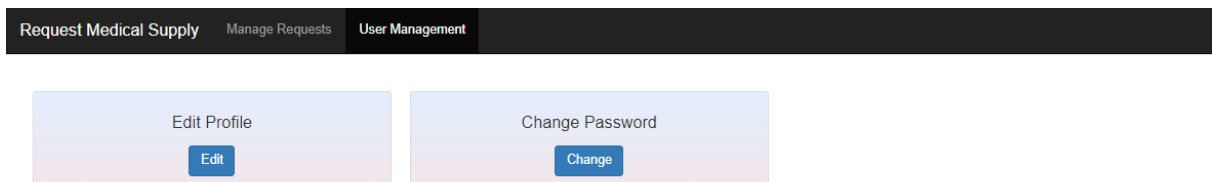


Figura 7.29: Aplicación para la petición de los usuarios "User Management"

7.8.2. Aplicación para la reasignación de las instituciones temporales

Para poder acceder al proceso explicado en la sección 7.6.3, el Programme Manager debe acceder a esta aplicación (ver Figura 7.30 creada que contiene un enlace que le permite acceder a esta herramienta y dar inicio al proceso en cuestión).

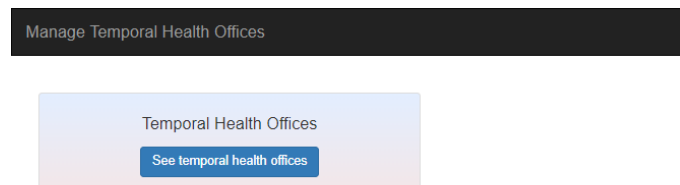


Figura 7.30: Aplicación para acceder a las Instituciones temporales

7.8.3. Aplicación para la consulta de todas las peticiones

Por otro lado, para que el Programme Manager pueda acceder al proceso que le muestra todas las peticiones, este tiene otra aplicación (ver Figura 7.31) donde podrá acceder al proceso en cuestión que le muestra esto.

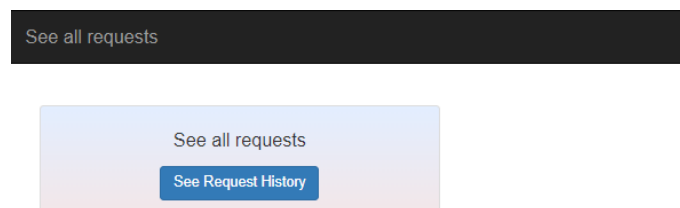


Figura 7.31: Aplicación para acceder al proceso de consulta de las peticiones

7.9. Testeo

Bonita Studio proporciona características para facilitar el testing que permite al desarrollador probar todo el proceso ejecutándolo o depurándolo. Al depurar un proceso, las tareas que necesitan ser probadas y sus conectores pueden ser seleccionadas o excluidas. Además, las pruebas se pueden realizar de forma más específica, reduciéndola a los conectores y expresiones. Este proyecto ha sido desarrollado utilizando las herramientas proporcionadas por Bonita BPM. Por lo tanto, la prueba de todos los procesos se realizó desde Bonita Studio en la máquina local. Todos los procesos fueron probados mientras se iban construyendo, por eso, cada vez que una nueva tarea, conector o expresión era creada, el proceso se ejecutó para verificar que los contratos de todas las tareas y el resto de los elementos del proceso estaban funcionando de la forma esperada para todos los casos posibles.

En el proyecto, el proceso "Request Process" fue el que más hubo que probar debido a la gran cantidad de posibilidades que ofrece, por un lado hubo que comprobar que funcionase bien tanto para los usuarios registrados como para los nuevo, por otro lado había que comprobar si el usuario quería cambiarse la institución o no, y por último según la institución las opciones se clasificaban en si creaba una nueva, elegía una de las temporales o escogía una de la lista "oficial" de Administration Units. Esto daba a pie a muchas formas de tratar la información y por lo tanto mucho tiempo haciendo pruebas.

Debido a la estructura de tareas BPM, fue fácil realizar prueba y los errores fueron detectados rápidamente. Finalmente, cuando cada proceso se completó, se realizó una prueba general para asegurar que las interacciones entre los procesos existentes en el sistema funcionaban como se esperaba.

Una vez que el proceso funcionaba bien, este estaba listo para ser implementado en el servidor. Los pasos seguidos para desplegar un proceso en el servidor, se explicarán en la siguiente sección. Además, en las distintas reuniones, se ha realizado una demostración para las partes interesadas de la OMS y puedan dar feedback de esta.

7.10. Despliegue del servidor

Para implementar un proyecto de Bonita se deben seguir algunos pasos, cada uno de ellos requiere autorización. Los diferentes usuarios y las autorizaciones que les fueron asignadas fueron descrito anteriormente en la sección [6.3.2](#). Los primeros pasos que deben seguirse cuando se despliega un proyecto de Bonita empiezan cargando el archivo XML de la Organización, que contiene los usuarios, grupos y roles existentes en el sistema, y el archivo .zip de BDM en el Portal. Para cargar el BDM, es necesario pausar los servicios BPM. Estas funcionalidades solo están disponibles para el Tenant Administrator (usuario detallado en sección [6.3.2](#)). Si no estamos implementando por primera vez o si no hemos modificado la Organización o el BDM este paso se puede omitir. Lo siguiente será importar todos los recursos (páginas, formularios, diseños, temas y Extensiones de API REST), procesos y aplicaciones que se han modificado o agregado desde la último despliegue en el Portal. Los recursos se cargan como archivos .zip, los procesos como archivos .bar creados desde el Bonita Studio, y las aplicaciones como archivos .xml. Estas funcionalidades están disponibles para los usuarios con el perfil de Administrador, el Tenant Administrator solo puede cargar los recursos, pero no puede cargar ni los procesos ni las aplicaciones.

Capítulo 8

Conclusiones

Las enfermedades tropicales desatendidas, representan una seria amenaza para la salud de todos los individuos y también a todos los países donde están presentes. Por eso, mejorando la distribución y el proceso de petición de suministros médicos es fundamental para facilitar su erradicación. De acuerdo con esto, los objetivos de la tesis fueron establecidos, como se detalla en la sección [1.4](#).

Este proyecto comenzó con el objetivo de crear una aplicación para adaptar así el proceso actual de petición de medicamentos, mejorando su eficiencia. Primero de todo se comenzó con una serie de pruebas y preparación del terreno en el que se iba a trabajar, lo ya establecido en el proyecto previo fue fundamental ya que los procesos se construyeron fácilmente y se probaron rápidamente desde Bonita Studio. A pesar de no utilizar la "Subscription Edition" de Bonita, la mayoría de los problemas que teníamos por esto pudieron ser eludidos y las limitaciones de la herramienta no causaron la modificación de las funcionalidades requeridas.

Una vez formalizado el protocolo e interiorizada la herramienta, se realizó la transformación digital utilizando la herramienta de Bonita y añadiendo al sistema una serie de procesos que anteriormente se realizaban usando hojas de cálculo, correos electrónicos o llamadas telefónicas. Mientras se realizaba, nos dimos cuenta de que no podíamos registrar usuarios desde dentro de Bonita y por lo tanto tuvimos que pensar una idea diferente para conseguir esto. Además, debido al hecho de que las herramientas BPM se combinan con otros para gestionar a todos los usuarios, el sistema es consciente del estado actual y quién está haciendo cada tarea. Esta funcionalidad

era muy importante debido a que los procesos requerían la interacción de varios actores, por lo que era necesario controlar lo que estaba pasando.

En conclusión, esta implementación ha automatizado la coordinación y asignación de tareas (es decir, asignándolas directamente al (los) actor (es) correspondiente (s), y también automatizando algunas de las tareas que anteriormente se realizaban manualmente (es decir, notificaciones por correo electrónico). Además, el sistema tiene otros beneficios como ser un punto central de coordinación, tener un sistema de seguimiento y almacenar todos los datos importantes y necesarios.

Además, todos los procesos fueron diseñados e implementados de tal manera que pueden ser adaptados a todas las enfermedades posibles. Así, la incorporación de nuevas enfermedades deben requerir modificaciones menores.

Finalmente, el último objetivo se centró en documentar los pasos relevantes seguidos. Se estableció un objetivo para acelerar la integración de los futuros desarrolladores que se unirán a la proyecto. La documentación detalla los pasos de configuración y los principales problemas detectados. En conclusión, los objetivos planeados para la tesis, que representan los siguientes pasos de un proyecto mucho mayor especificados en la sección 1.4 han sido conseguidos.

Finalmente, las competencias técnicas requeridas para completar el proyecto se detallan en Apéndice B.

Apéndice A

Símbolos BPM

A.1. Gateways

A.1.1. Parallel gateway

Cuando el flujo alcanza una Parallel gateway, si este se divide en dos caminos, estos dos son ejecutados a la vez.

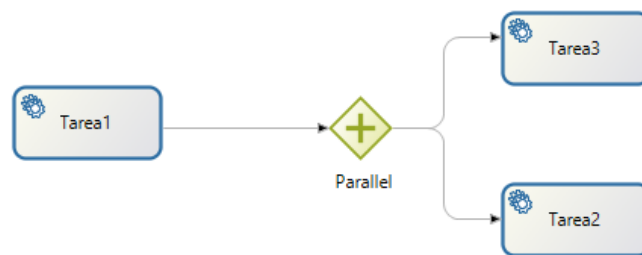


Figura A.1: Parallel gateway

A.1.2. Exclusive gateway

Cuando el flujo alcanza una Parallel gateway, si este se divide en dos caminos, solo uno es ejecutado.

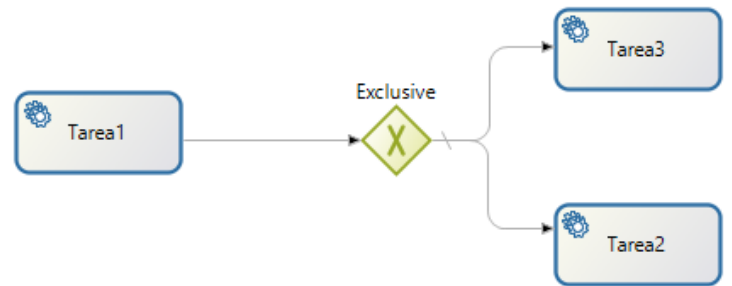


Figura A.2: Exclusive gateway

A.1.3. Inclusive gateway

Cuando el flujo alcanza una Inclusive gateway, este espera a que todos los otros flujos que puedan haber, lleguen. Cuando esto ocurre, se evalúan las diferentes salidas para saber por donde seguir.



Figura A.3: Inclusive gateway

A.2. Tasks

A.2.1. Human task

Las Human task se vinculan a un actor y son completadas cuando este accede al formulario y lo envía.

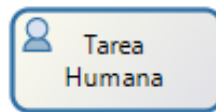


Figura A.4: Human task

A.2.2. Service task

Las Service Task, son tareas que se completan automáticamente por el sistema.

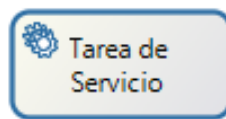


Figura A.5: Service task

A.2.3. Task with connector

Cuando una tarea tiene un conector, esto se indica con el icono de un enchufe.

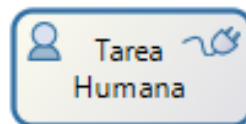


Figura A.6: Task with connector

A.3. Activities

A.3.1. Call activity

Una Call activity llama a un proceso, esto se puede hacer de tres formas: una llamada simple, una llamada de forma repetida y paralela u otra llamada de forma repetida pero secuencial.

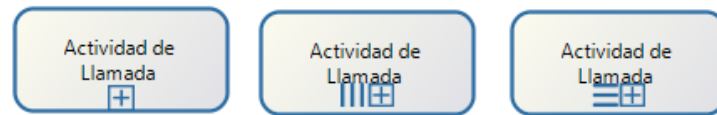


Figura A.7: Call activity

A.4. Events

A.4.1. Start event

Marca el inicio de un flujo de un proceso. Es representado mediante un círculo verde.

A.4.2. End event

Marca el final de un flujo de un proceso. Es representado mediante un círculo rojo

A.4.3. Non-interrupting timer boundary event

Cuando el temporizador que tiene asociado llega a cero, lanza un flujo que realiza una o más tareas.

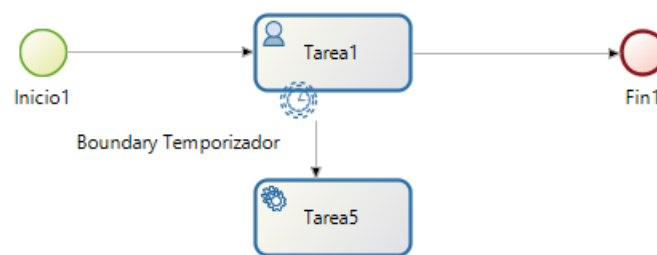


Figura A.8: Events

Apéndice B

Competencias Técnicas de Ingeniería del Software

B.1. Competencias Técnicas

A continuación se van a justificar las competencias de ingeniería del software elegidas para realizar en este proyecto:

- CES1.1: Desarrollar, mantener y evaluar sistemas y servicios software complejos y / o críticos. [En profundidad]

Esta competencia coincide con el objetivo principal de este proyecto: desarrollar una aplicación web, y por lo tanto está completamente cubierto por el proyecto

- CES1.5: Especificar, diseñar, implementar y evaluar bases de datos. [Bastante].

Esta competencia está bien cubierta debido a que la herramienta (Bonita BPM) solo incluye una base de datos predeterminada que no se puede utilizar en producción. Por lo tanto, la especificación, diseño, implementación y evaluación de una nueva base de datos desde cero

- CES1.7: Controlar la calidad y diseñar pruebas en la producción de software. [En profundidad]

El sistema debe proporcionar calidad y los procesos deben completarse con éxito. Es por eso que todos los flujos son probados a fondo para garantizar su efectividad.

- CES1.9: Demostrar comprensión en la gestión y gobierno de los sistemas software. [Bastante]

Dado que el proyecto parte del aprendizaje de unas herramientas y llega a la implementación y pruebas, se consigue bastante la comprensión en la gestión y el gobierno del sistema

.

- CES2.1: Definir y gestionar los requisitos de un sistema software. [En profundidad]

Durante la primera fase del proyecto, los requisitos fueron cuidadosamente estudiados y se tuvieron en cuenta para encontrar la solución mas apta para la conversión del proceso a una aplicación.

- CES2.2: Diseñar soluciones apropiadas en uno o más dominios de aplicación, utilizando métodos de ingeniería del software que integren aspectos éticos, sociales, legales y económicos. [Bastante]

El proyecto contribuye proporcionando una herramienta que ayudará a la OMS a lograr su objetivo: Poder controlar y distribuir de una forma mucho más eficiente los diferentes medicamentos al resto del mundo según reciban las diferentes peticiones. En este sistema estaremos cubriendo aspectos éticos, sociales, legales y económicos.

Bibliografía

- [1] AuraPortal. <https://www.auraportal.com/es/destacados/que-es-bpm/>
- [2] KissFlow (2019).
<https://kissflow.com/bpm/business-process-management-overview/>
- [3] BonitaSoft. <https://www.bonitasoft.com/>
- [4] API BonitaSoft. <https://documentation.bonitasoft.com/bonita/7.8/>
- [5] BPM Social Media. <https://www.bpmsocialmedia.com/>
- [6] Redmine. <https://www.redmine.org/>
- [7] Deloitte. María Lahoz.
<https://www2.deloitte.com/es/es/pages/technology/articles/business-process-management-software-componentes-ciclo-vida.html>
- [8] ¿Qué es la metodología 'agile'?. Maria Tena. <https://www.bbva.com/es/metodologia-agile-la-revolucion-las-formas-trabajo/>,
Noviembre 2018
- [9] Áreas de conocimiento del BPM.
<https://www.evaluandosoftware.com/areas-conocimiento-del-bpm/>
- [10] Elevate Efficiency with Business Process Management (BPM) Applications. Nisha Achuthan (2018). <https://kissflow.com/bpm/bpm-application/>

- [11] When do you really need custom application development?. Miguel Valdes Faura.
[https://techbeacon.com/app-dev-testing/
when-do-you-really-need-custom-application-development](https://techbeacon.com/app-dev-testing/when-do-you-really-need-custom-application-development)
- [12] BPM SPAIN. <https://www.bpm-spain.com/>
- [13] Business Process Definition.
<https://www.appian.com/bpm/definition-of-a-business-process/>
- [14] Business Process.
<http://www.pnmsoft.com/resources/bpm-tutorial/business-process/>
- [15] Automatiza y modeliza tus procesos y optimiza al máximo el tiempo de trabajadores y colaboradores. [https:
//www.inycom.es/soluciones-tecnologicas-business-process-management-bpm](https://www.inycom.es/soluciones-tecnologicas-business-process-management-bpm)
- [16] About TortoiseSVN. <https://tortoisesvn.net/about.html>
- [17] The Flowable Project. <https://www.flowable.org/>
- [18] Comparing and Contrasting Open Source BPM Projects. Andrew Bonham (2016)
[https://medium.com/capital-one-tech/
comparing-and-contrasting-open-source-bpm-projects-196833f23391](https://medium.com/capital-one-tech/comparing-and-contrasting-open-source-bpm-projects-196833f23391)
- [19] PagePersonnel(2017). [https://www.michaelpage.es/sites/michaelpage.es/files/
MP_SPA_ON_ER_IT_03052017.pdf](https://www.michaelpage.es/sites/michaelpage.es/files/MP_SPA_ON_ER_IT_03052017.pdf)
- [20] Thesis TFG. BPM implementation for the worldwide distribution of Azithromycin for Yaws eradication(2018). Clara Gaset. [https://upcommons.upc.edu/bitstream/handle/
2117/121976/136043.pdf?sequence=1&isAllowed=y](https://upcommons.upc.edu/bitstream/handle/2117/121976/136043.pdf?sequence=1&isAllowed=y)

